International Institute for Applied Systems Analysis
IIASA www.iiasa.ac.at

Address: IIASA, Schlossplatz 1, A-2361 Laxenburg, Austria

Email: permissions@iiasa.ac.at

Young Scientists Summer Program

Harmonizing different time-scale decisions in renewable energy generation plants

[zixuan_zhang@mail.ecust.edu.cn]

Approved by:

Mentor(s): Marek Makowski, Julian Hunt
Program: Energy, Climate, and Environment
Date: 21 April 2023

This report represents the work completed by the author during the IIASA Young Scientists Summer Program (YSSP) with approval from the YSSP mentor.

It was finished by 21 April 2023 and has not been altered or revised since.

Mentor signature:

The report was approved by the email communication.

Contents

Abstract 3

About the author 4

Acknowledgments 4

1 Introduction 5

2 Method for selecting time slices 6
- 2.1 Specification of the data 6
- 2.2 Data preprocessing 7
  - 2.2.1 Data normalization 7
  - 2.2.2 Preparing data elements for multi-region models 7
- 2.3 Approach to defining aggregated data representation 7
- 2.4 Definitions of the used terms 7
- 2.5 Overview of the approach 8
- 2.6 First stage: first stream 10
  - 2.6.1 Split of one-hour slices into seasons 10
  - 2.6.2 Sorted values of attributes 10
  - 2.6.3 Defining ranges of attribute values 10
- 2.7 First stage: second stream 11
  - 2.7.1 Aggregation of hourly data 11
  - 2.7.2 Split aggregated data into subsets corresponding to four seasons 11
  - 2.7.3 Data defining artificial weeks 12
- 2.8 Second stage 13

3 Results for selecting time slices 15

4 Operational model framework 17
- 4.1 Indexing structure 17
  - 4.1.1 Decision variables 17
  - 4.1.2 Outcome variables 18
  - 4.1.3 Exogenous variables 18
  - 4.1.4 State variables 18
  - 4.1.5 Auxiliary variables 18
- 4.2 Parameters 19
- 4.3 Relations 19

5 Conclusion 20

[0] ZVR 524808900

The authors gratefully acknowledge funding from the National Natural Science Foundation of China's IIASA Young Scientists Summer Program 2022' (72211540122).

Abstract

Planning and operation of the renewable energy system (RES) face challenges of high variability and uncertainty. Decisions that consider the evolution of the energy system over multiple decades under the uncertainty of parameters tend to neglect short-term variability. In contrast, operation decisions that deal with the short-term variability of RES, such as balancing hourly energy generation with requirement outputs, would neglect its long-term evolution. Thus, the harmonization decision of different time scales is a key issue in modeling RES. This report presents the progress in developing a model to support the long-term planning and short-term operation of RES. The RES includes renewable energy inputs and storage systems with electrolyzers, hydrogen tanks, and fuel cells. The report includes two main parts. The first part focuses on connecting the operation and the long-term planning of RES by using aggregated data in the MESSAGEix framework. The approach developed by Dr. Julian Hunt and Dr. Behnam Zakeri for dealing with and selecting aggregated data is presented in this part. We also present the case results for the approach. The second part focuses on establishing a separate model focus on the operation of a given RES. This model is the basement operational model for connecting with the planning model of RES. The presented version of the model deals with a given RES, in which hourly operation decisions will be made to balance the variability of generated energy with the requirements of stable outputs. The current version of the model is still at an early stage of development. Future versions can support both planning RES expansion and multi-product analysis involving hydrogen.

About the author

Zixuan Zhang is currently a Ph.D. student at the center of energy economics and environmental management at East China University of Science and Technology. Her current research focus is harmonizing different time-scale decisions in energy system analysis. She has experience with Input-Output analysis, and her main interests are energy system modeling, energy storage, energy system resilience, and environmental economics. (Contact: zixuan_zhang@mail.ecust.edu.cn)

Acknowledgments

It's my great honor to have the chance to join the YSSP. The summertime I spent at IIASA is grateful and unique. The discussions with different people are informative with various exciting experiences and ideas. 'Think globally at IIASA' is the greatest thing I have learned.

Through my participation in the YSSP program, I was able to make significant improvements and advancements in my long-term research endeavors. This research not only contributes to the activities of my team but also serves as a pivotal component of my ongoing Ph.D. studies.

I've learned new technics and assimilated diverse perspectives and innovative ideas during IIASA. A deep understanding of a specific research method is the beginning of getting familiar with problem-solving in this research field. The concept of SMS has deeply inspired my research. These works have become a cornerstone of my academic pursuits.

I would first like to thank my supervisor Prof. Marek Makowski and Dr. Julian Hunt, for their guidance and patience through each stage of the process. Their insightful and meticulous feedback pushed me to sharpen my thoughts. Through the informative discussion, I have learned not only how to overcome a big problem, but also how to express ideas effectively. Their Every time I practice, I remember their guidance and encouragement. I would also like to thank Dr. Behnam Zakeri, for his time and support in helping me understand the tools I needed. I would like to thank Dr. Takuya Hara, from whom I learn how to be motivated.

Then I would like to thank the friends I met during YSSP, for their different sense of research and wonderful experiences in life. Their stories inspire me to think more and work hard for creating better experiences in both research and life. Also thanks to Tanja Huber, Aleksandra Cofala, Pat Wagner, Margit Roth-Ahra, and other staff who support me in completing and enjoying the journey in IIASA.

Last but not the least, thanks to my country China, and my supervisors Prof. Tieju Ma, Yadong Yu, and Hongtao Ren who support me to experience more in research life.

The valuable experience gained during my YSSP research project will enable me to make substantial strides forward in my long-term research objectives.

1 Introduction

The high proportion of unstable renewable energy generation, such as wind and photovoltaic power, into the grid, puts forward higher requirements for the stability of the power system. In 2020, Sweden generated 20% of its electricity from wind and photovoltaic, followed by German, 18%. While in China, wind and photovoltaic power generation accounts for 7.8% and 3.9% of total electricity generation in 2021, respectively [1]. Although the proportion of renewable energy generation is still at a lower level in China, it is foreseeable that China will have over 50% electricity from wind and photovoltaic power generation in 2030, according to its 'carbon peaking' and 'carbon neutrality targets. wind and photovoltaic power will increasingly participate in power generation and gain a dominant position in the power system. However, wind and photovoltaic are unstable power suppliers. The fluctuation for wind power can be up to 80% within a day. The short-term forecasts of wind power generation can be accurate within a day, but the accuracy of long-term forecasting is poor. The utilization of solar energy is diurnal and seasonal variation [1]. For this, it is challenging to provide stable power output. The flexibility and uncertainty of this renewable energy generation have pressured the power system to balance supply and demand. Then, the importance of energy storage is becoming increasingly prominent. Energy storage equipment, such as batteries, is mandatory to construct with at least 2 hours of storage for new wind farms in China. However, such measures can only support short-term adjustments, there are still challenges to the long-term stable power supply of the wind farm.

Hydrogen energy storage (HES) systems can supplement renewable energy sources to overcome the challenges associated with higher penetrations of wind-based electricity both in the short-term and long-term [2]. The hydrogen energy storage system contains hydrogen production, hydrogen storage, and power converting from hydrogen, HES system fills out the lack of batteries which only offers short-term flexibility services for the electricity network. The overproduced power can be converted into green hydrogen and stored as compressed gas or liquid hydrogen for a while. Then, the hydrogen can be converted back into power for supply. There's almost no loss during the long-term storage of hydrogen [3]. Moreover, the maturation of technology in hydrogen energy storage systems now can support hydrogen production from unstable input power. So, there might be a potential that a wind-hydrogen system, which we refer to as a wind farm with a co-located HES system, can provide a stable power supply like traditional thermal power plants.

Harmonization of different time scales is a key research and model-implementation issue. Investment planning of RES can last for decades. It considers the influence of planning time, lifespan, and discount rate. The operation of RES focuses on decisions of storage devices to cope with the variability of the generated energy. The decisions can change in the short-term, such as by hourly, or in the medium-term, such as seasonal. In planning and operations of RES, the decision-makers have to consider trade-offs between reachable goals for conflicting objectives, such as total costs, investment costs, meeting the requirements for the produced energy. Therefore, the model-based decision support will be based on tools and methods for Multiple-Criteria Model Analysis (MCMA).

There have been two streams for harmonization of different time scales in modeling renewable power systems [4]. One stream is called direct integration. Direct integration is always applied in energy system optimization models. These models can provide a macroscopic and

[1] Data from https://www.baogaoting.com/info/219153

comprehensive description of the evolution of the energy system for multi-years [5]. When considering inter-temporal transactions, a finer level of time representation (such as seasons and weeks) will help the model reflect reality, especially in models dealing with different time period energy storage problems [6]. By using this direct integration, sub-annual variations, such as daily or seasonal, in demand and supply are represented by time-slices. Data contained in each time-slice will enhance the accurateness of the long-term models (e.g., Models built on MESSAGE modeling framework). The key issue for this method is how to select the represented time-slices [7].

The other stream is called soft-link model coupling methodologies, where a soft-link between the long-term planning and short-term operating models, which only contain a limited level of temporal and technical detail, is established.

The purpose of the research is to develop a model to harmonize decisions regarding long-term planning and operations of RES. The work conducted during YSSP explored two streams of approaches to harmonize decisions across different time scales. Section 2 summarizes the time-slices selecting approach developed by Dr. Julian Hunt and Dr. Behnam Zakeri. This approach is the basement for integrating short-term operating conditions of storage into long-term planning model. Section 3 presents the results of the time-slices selecting approach. Section 4 shows an initial symbolic model specification (SMS) of an operational model. The reported work has provided a foundation for future analysis.

2 Method for selecting time slices

The processes of the method presented in this section reflect the author's comprehension of the approach developed by Dr. Julian Hunt and Dr. Behnam Zakeri. Those were built during numerous discussions, email exchanges, and analyses of the provided Python code.

2.1 Specification of the data

The following data are used for illustration of the *time-slices* prototype:

- Hourly data for one year.
- The data provided in XLS are converted into the Python data-frame, here presented as matrix H; its rows and columns are indexed by $i \in I$ and $j \in J$, respectively. Therefore,

$$H = h[i][j], i \in I, j \in J \tag{1}$$

where $h[i][j]$ are elements of data matrix H. Members set I index the records; therefore, $I = \{0, 1, \ldots, 8759\}$.[2] Set J is implicitly defined by eq. (4) below.

Equivalent notation of definition (1):

$$H = \{h_{ij}\}, \qquad (\text{or } \; H = h_{i,j}), \qquad i \in I, j \in J$$

- Each data record (a row of H) consists of:
  - ⋆ time-stamp composed of two values: calendar day and hour,

[2] 365 days * 24 hrs/day = 8760. If the data is provided for a leap year, then these numbers are increased by 24. For simplifying the presentation we assume here a year of 365 days.

⋆ a given number of values considered for each time slot; the values are organized in vectors denoted as $\boldsymbol{v}$, i.e.,

$$\boldsymbol{v} = \{v_k\}, k \in K, \tag{2}$$

where indices k identify elements of the data provided for every day and hour.

For the problem considered in this note these values represent: solar radiation, wind, demand, hydro, respectively. Therefore,

$$\boldsymbol{v} = \{solar, wind, demand, hydro\}. \tag{3}$$

Components of vector $\boldsymbol{v}$ can easily be modified for various compositions of the hourly data; such modifications will not require any changes of the applied data structure.

- Therefore the i-th data record can be presented in two equivalent forms, each suitable for a particular context:

$$\boldsymbol{h}[i] = \{day, hour, \boldsymbol{v}\} = \{day, hour, \{solar, wind, demand, hydro\}\}, \qquad i \in I. \tag{4}$$

2.2 Data preprocessing

The described approach does not require any preprocessing of the provided data. Therefore we only briefly mention two types of preprocessing that is required for either other data sources or for extensions of the approach to multi-regions.

2.2.1 Data normalization

Values of all provided data series have been normalized to the range [0, 1].

2.2.2 Preparing data elements for multi-region models

In order to handle day-related variability when dealing with energy transfers between regions in different time-zones one has to harmonize data using either the local or universal time. This issue will be considered at a later research stage.

2.3 Approach to defining aggregated data representation

The overall aim of the summarized approach is to describe the algorithm for finding an aggregated representation of original detailed trajectories of data characterizing energy produced by diverse VREs (Variable Renewable Energy) sources and the energy demand.

2.4 Definitions of the used terms

The following terms are used in the note:

- Time-slice represents data within a period. The data is organized into a corresponding matrix containing (possibly processed) values of the original data-vector $\boldsymbol{v}$.
- Four-hour time-slice: a matrix, with each row contains data representing 4 consecutive hours, by four values, each of them corresponding to one component of original data-vector $\boldsymbol{v}$.
- Day: natural day composed of 24 one-hour time-slices denoted by 0:00 through 23:00.

- aDay: artificial day consisting of given number time-slices, representing 24 one-hour time-slices.
- Week: an ordered set of 42 four-hour[3] consecutive time-slices.
- Season: an ordered set of 2190 one-hour time-slices or 547 four-hour time-slices.
- P is an auxiliary data structures used for defining $\{$D1, D2$\}$.
- Load duration curve: k type of data with a descending order, which sorted based on the value of k type of data from the highest to the lowest.[4]

2.5 Overview of the approach

The goal of the approach is to find a representation of the original annual hourly data by eight objects (four pairs of $\{$D1, D2$\}$; each pair representing original data in one of four seasons), each of them composed of six time-slices. Figure 1 illustrates the flows between basic elements of the algorithm; these elements are defined in detail in subsequent sections.

The process runs for each of the four seasons independently. Therefore, for the sake of brevity, after illustrating the split of data between seasons, we show the flow only for the first season.

In the first stage the original is processed in two parallel streams in order to prepare data structures needed for defining P:

- In the first stream the original data (defined for one-hour time-slices) is processed in the following steps:
  1. The data is divided into seasons, see Section 2.6.1
  2. For each season, matrix C is created to store sorted values of attributes v of H, see Section 2.6.2
  3. For each season matrix R composed of the the maximum and minimum values of data attributes are defined, see Section 2.6.3
- The second stream deals with the data aggregated into four-hour time-slices through the following steps:
  1. The one-hour time-slices are aggregated into four-hour time-slices, see Section 2.7.1
  2. The aggregated data is divided into seasons, see Section 2.7.2.
  3. For each season the data is split into aWeeks, see Section 2.7.3.

After completing the first stage the two data objects are available for further processing:

- From the first stream: matrix R containing maximum of minimum values of each of the original data.
- From the second stream: data structures for each of 13 aWeeks (artificial weeks for each season) containing six 4-hour time-slices for each of the aDays of the corresponding aWeek.

The second stage consists of the iterative procedure defined in Section 2.8 below.

[3] Six slices for each of seven consecutive days.

[4] The load duration curve is defined as the curve between the load and time in which the ordinates representing the load, plotted in the order of decreasing magnitude. In our data, there are four kinds of load: solar, wind, hydro and demand. The load duration curve reflects the duration time of one load at different production or demand levels.

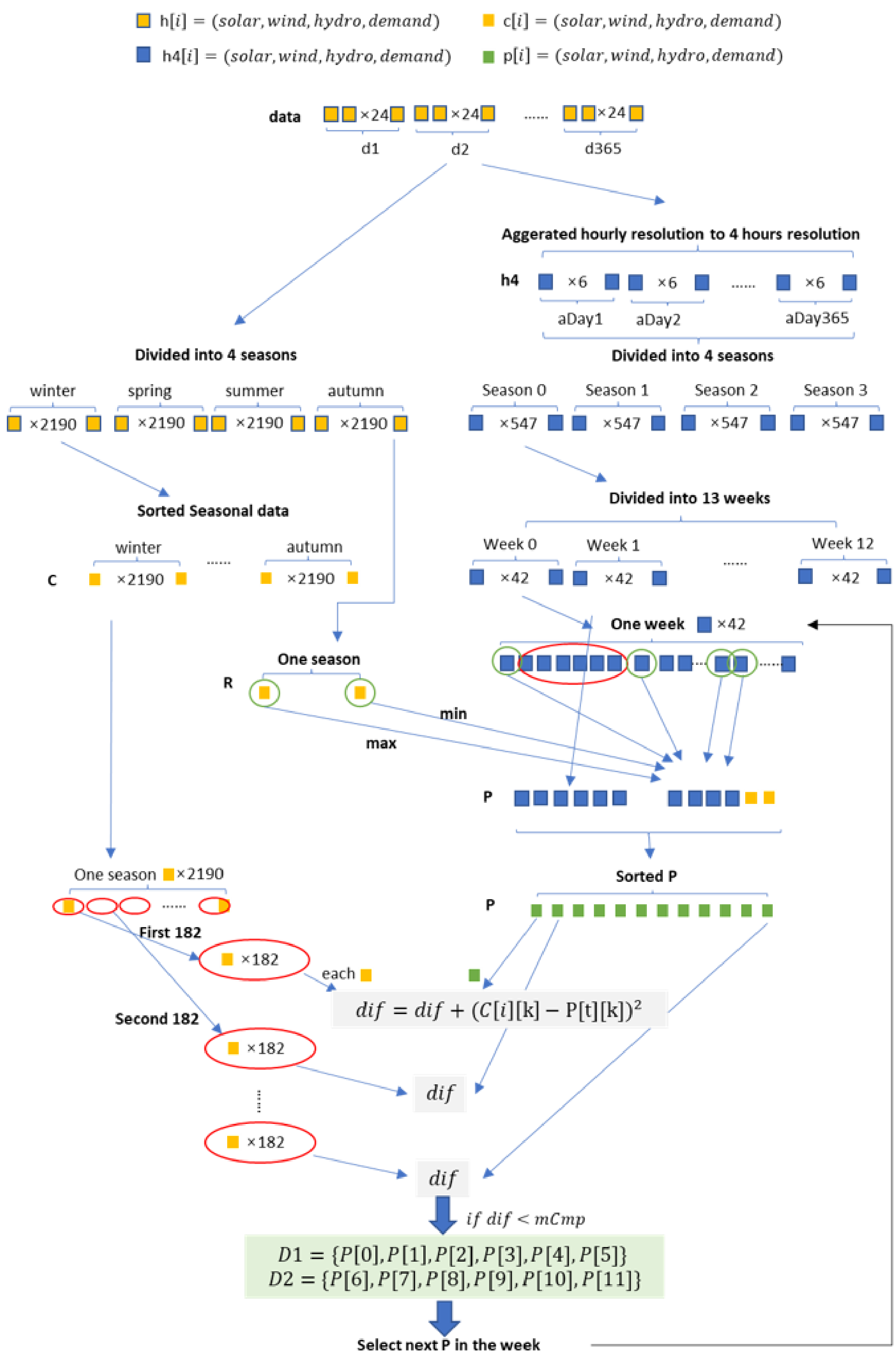

Figure 1: Overview of the algorithm.

2.6 First stage: first stream

2.6.1 Split of one-hour slices into seasons

One-hour time-slices stored in H are split into four subsets H_s, respectively, each subset corresponding to a calendar season. This is done by defining subsets of the corresponding row-indexing set I into four (not overlapping) subsets, i.e.,

$$I = \{I_s\}, \qquad s \in S, \quad S = \{0, 1, 2, 3\}, \tag{5}$$

where members of S index annual seasons (Winter, Spring, Summer, Autumn). The subsets I_s are defined based on calendar days as follows:

- $I_0 = \{0, 1, \ldots, 2189\}$,
- $I_1 = \{2190, 2191, \ldots, 4379\}$,
- $I_2 = \{4380, 4381, \ldots, 6569\}$,
- $I_3 = \{6570, 6571, \ldots, 8759\}$.

2.6.2 Sorted values of attributes

Create auxiliary matrix C composed by sorted values of attributes $\boldsymbol{v}$ stored in H. Pseudo-code of such a mapping is shown in Fig. 2.

```
for s ∈ S do
    for k ∈ K do
        for i ∈ I_s do                      ▷ store values in auxiliary vector x.
            x[i] = H[i].v[k]
        end for
        x = sort(x)                         ▷ Sort the vector by decreasing values.
        for i ∈ I_s do                      ▷ store values in auxiliary vector x.
            C[s][i][k] = x[i]               ▷ Create k-th column of C by sorted values.
        end for
    end for
end for
```

Figure 2: Pseudo-code for $H \to C$ mapping.

2.6.3 Defining ranges of attribute values

The ranges of attribute values in matrix H are stored in matrix R composed (for each season) of two rows, containing minimum and maximum values of each attribute, respectively. The corresponding pseudo-code is presented in Figure 3.

for $s \in S$ **do**
  for $k \in K$ **do**
    for $i \in I_s$ **do**
      $x[i] = H[i].v[k])$
    end for
    $R[s][k][low] = arg \min_{i \in I_s} \boldsymbol{x}$
    $R[s][k][upp] = arg \max_{i \in I_s} \boldsymbol{x}$
  end for
end for

Figure 3: Calculation of the seasonal ranges of values of attributes $\boldsymbol{v}$.

2.7 First stage: second stream

2.7.1 Aggregation of hourly data

In this step four consecutive one-hour time-slices are aggregated into one corresponding four-hour slices and stored in matrix is denoted as H4. In order to distinguish indexing (by $i \in I$) one-hour time-slices of H, we index four-hour time-slices of $H4$ by $t \in T$, where $T = \{0, 1, \ldots, 2189\}$.[5] Therefore elements of $H4$ are denoted by $h4[t][j], t \in T, j \in J$. As the result of such an aggregation the values $\boldsymbol{v}$ are defined for each t-th time-slice. Rows of $H4$ have the same structure as rows of H, see (4), namely:

$$\boldsymbol{h}4[t] = \{day, hour, \boldsymbol{v}\} = \{day, hour, \{solar, wind, demand, hydro\}\}, \qquad t \in T. \tag{6}$$

The following algorithm (outlined by the corresponding pseudo-code shown in Figure 4) defines mapping $H \rightarrow H4$. We explain here the key elements of this mapping:

- Record $h4[0]$ is set to be equal to $h[0]$, representing Jan 1, 0:00.
- All other data records of $H4$ are defined by aggregations of four consecutive records of H, starting with $h[1]$, which corresponds to Jan 1, 1:00.
- The records of $H4$ have the time-stamps of the last of each four aggregated records of H.
- The last three records (corresponding to hours 21:00, 22:00, 23:00) are aggregated with the first record of the next day (0:00).
- The last three records of Dec 31 in H are ignored.

2.7.2 Split aggregated data into subsets corresponding to four seasons

In order to provide convenient access to subsets of time-slices belonging to each calendar season, the set of indices $t \in T$ of $H4$ is split into four subsets T_s:

$$T = \{T_s\}, \qquad s \in S, \quad S = \{0, 1, 2, 3\}, \tag{7}$$

where the subsets are defined based on calendar days as follows:

- $T_0 = \{0, 1, \ldots, 546\}$,

[5] 365 days * 6 four-hour time-slices/day = 2190.

```
i = 0                                   ▷ current time-slice of H
for t ∈ T do
    if i == 0 then        ▷ First time-slice of H is copied (without aggregation) to H4.
        h4[t].day = h[i].day
        h4[t].hour = h[i].hour
        for k ∈ K do
            h4[t].v[k] = h[i].v[k]
        end for
        i = i + 1
    else                  ▷ All but first four subsequent time-slices of H are aggregated.
        h4[t].day = h[i + 3].day
        h4[t].hour = h[i + 3].hour
        for k ∈ K do
            h4[t].v[k] = (Σ_{p=i}^{i+3} h[p].v[k])/4
        end for
        i = i + 4
    end if
end for
```

Figure 4: Pseudo-code for $H \rightarrow H4$ mapping.

- $T_1 = \{547, 548, \ldots, 1093\}$,
- $T_2 = \{1094, 1095, \ldots, 1640\}$,
- $T_3 = \{1641, 1642, \ldots, 2188\}$.

2.7.3 Data defining artificial weeks

We consider artificial weeks (aWeek)[6] each composed of seven consecutive days of the $H4$. Each day of $H4$ is represented by 6 four-hour time-slices; therefore, data for each aWeek consists of data (values of $\boldsymbol{v}$) of 42 consecutive time slices of $H4$. For each season 13 weeks are defined; thus we have 52 weeks in a year representing 364 days, i.e., 2184 four-hour time slices. Thus, five time-slices of $H4$ that don't fit any week are ignored.

The data (values of $\boldsymbol{v}$) corresponding to each week are accessed through subsets of consecutive time-slice defined for each season. Weeks are indexed by $w \in W,\ W = \{0, 1, \ldots, 12\}$ (same for all seasons) and two subsets of slices are denoted by T_{sw} and $T2_{sw},\ s \in S, w \in [0, 12]$.

[6] We call them artificial weeks because they do not correspond to calendar weeks.

2.8 Second stage

In the second stage we define for each season two artificial days, named $D1[s]$ and $D[s]$, each of them composed of six consecutive 4-hour time slices; each time-slices consists of values of four attributes (elements of $\boldsymbol{v}$). Values in $D1[s]$ and $D2[s]$ are defined using the algorithm shown in Fig. 5, which uses the data structures defined above, specifically matrices C, R, and $H4$ and the corresponding indexing sets. Additionally, we define weights $\boldsymbol{q}$ for each data-type of $\boldsymbol{v}$:

$$\boldsymbol{q} = \{q_k \geq 0,\ k \in K\} : \sum_{k \in K} q_k = 1. \tag{8}$$

Discussion of the specification of $\boldsymbol{q}$ is beyond the scope of this note.

We summarize now the basic elements of the algorithm presented in Fig. 5:

1. Each season is considered independent of all other seasons.
2. We use time-slices of $H4$ in subsets of each consecutive week of each season.
3. The time-slices for each week are selected in an iterative way, and placed in the auxiliary structure P, which is composed of 12 slices. For each selection, the following is observed:
   - First six consecutive slices are selected. For the first iterations, this subset starts with the first slice of the week. For the next iteration, the starting slice is the next one after the first slice of the previous selection. There are 42 slices each week. Therefore, the first slice of the last selection is the 37th slice of the week. Thus, there are 36 selections for each week. Such six slices are placed in P as the top six slices.
   - Next, four other slices are selected from the same week. In order to provide for repeatability of the results, we select the other four slices in the following way. We select the next or previous (immediately after or before the above selected six slices) four slices, for the first or second 18 selections/iterations. These slices are defined in $T2_{sw}$.
   - The last two slices of P are filled by low- or upper-bounds stored in the auxiliary matrix R.

   Note that only values of $\boldsymbol{v}$ are stored in P, i.e., no time-stamps are stored there.
4. The columns of P (corresponding to attributes of the corresponding components of $\boldsymbol{v}$ are sorted in descending order.
5. A difference between P (composed of 12 slices) and C is computed as defined in the pseudo-code. Note the use of weights $\boldsymbol{q}$ defined by (8).
6. If such a difference is the smallest amongst all computed in the season, then the slices of P define the $D1$ and $D2$ in the way shown in the pseudo-code.

```
mCmp = 1e + 9
iCmp = 182                                  ▷ Number of C slices used for comparisons with each P slice.
for s ∈ S do
    for w ∈ W do
        for t ∈ T_sw do                      ▷ Loop for the slice that starts a subset of 6 time-slices.
            for m ∈ [0, 5] do                ▷ Loop of 6 selected slices.
                for k ∈ K do
                    P[m][k] = h4[t + m].v[k]
                end for
            end for
        end for
        for t ∈ T2_sw do                     ▷ Loop for four other selected slices.
            for m ∈ [0, 3] do                ▷ Loop of four lower P slices.
                for k ∈ K do
                    P[m + 6][k] = h4[t + m].v[k]
                end for
            end for
        end for
        P[10][k] = R[s][k][upp]
        P[11][k] = R[s][k][low]
        for k ∈ K do                         ▷ Sort each column of P in by decreasing values.
            x = P[k]
            x = sort(x)                      ▷ Sort the vector by decreasing values.
            P[k] = x                         ▷ Replace the column in P by its sorted values.
        end for
        cmp = 0.
        for t ∈ [0, 11] do                   ▷ Time-slices in P.
            for k ∈ K do
                for m ∈ [0, iCmp] do         ▷ Comparisons with iCmp consecutive C slices.
                    cmp = cmp + q[k] · (C[T_s][0] + m]][k] − P[t][k])^2
                end for
            end for
        end for
        if cmp < mCmp then
            for t ∈ [0, 5] do                ▷ Time-slices in D1 and D2.
                for k ∈ K do
                    D1[s][t][k] = P[t][k]
                    D2[s][t][k] = P[t + 6][k]
                end for
            end for
            mCmp = cmp
        end if
    end for
end for
```

Figure 5: Pseudo-code for defining representative artificial days D1 and D2.

3 Results for selecting time slices

The key objective of the selection method outlined above is to identify representative time slices that effectively capture the trends in renewable energy generation and demand throughout the year. In this section, results obtained from the aforementioned approach are presented. We selected 48-time slices, each containing three types of hourly data (solar, wind, and demand) to represent the trend of hourly changes in the west of China. The red line in Fig.6 represents the original data for 8760 hours, while the blue line represents the selected 48-time slices, as shown in the figure.

Fig.6 presents a comparison of the annual load duration curves between the original data (8760 hours) and a selection of 48-time slices.

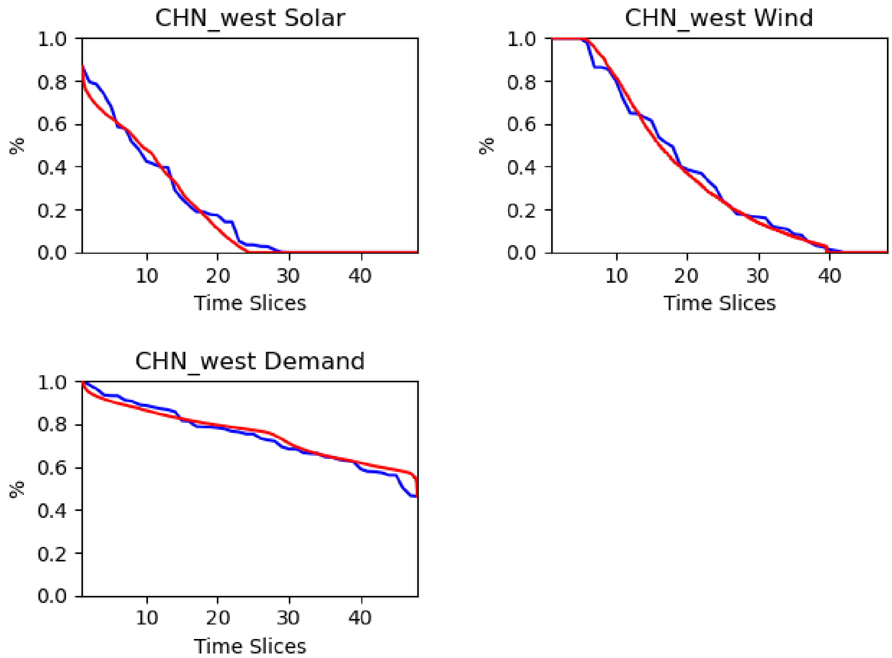

Figure 6: Annual results of 48-time slices and its comparison with original data.

Fig.7, Fig.8, and Fig.9 illustrate a seasonal load duration curve comparison between the original data and the selected 48-time slices, separately for solar, wind, and demand.

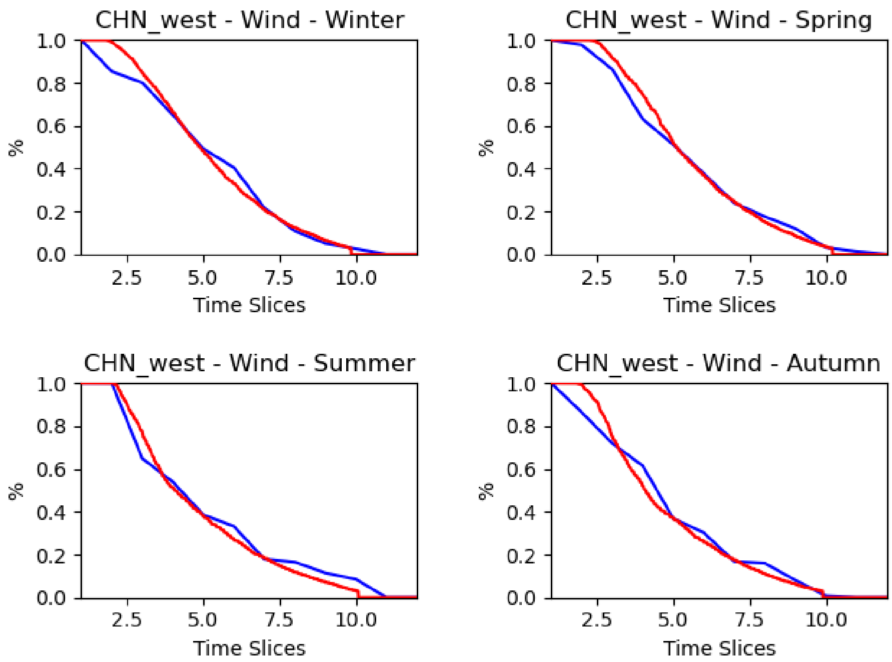

Figure 7: Seasonal wind results of 48-time slices and its comparison with original data.

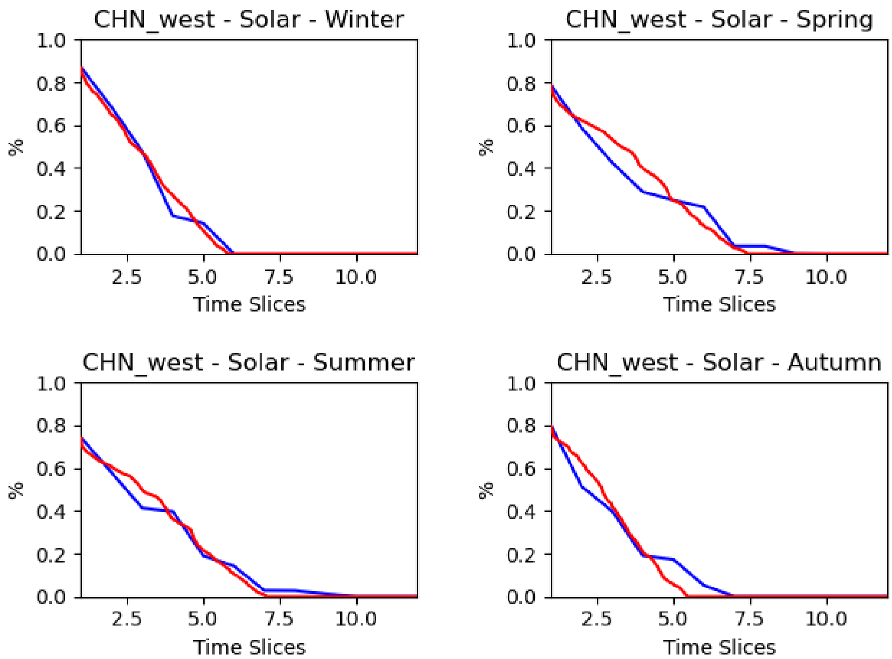

Figure 8: Seasonal solar results of 48-time slices and its comparison with original data.

It is worth noting that the selected time slices closely align with the trend of the original data, whether observed over the course of a year or on a seasonal basis. This demonstrates the effectiveness of our approach in capturing the key trends in renewable energy generation and demand, providing valuable insights for further analysis and modeling.

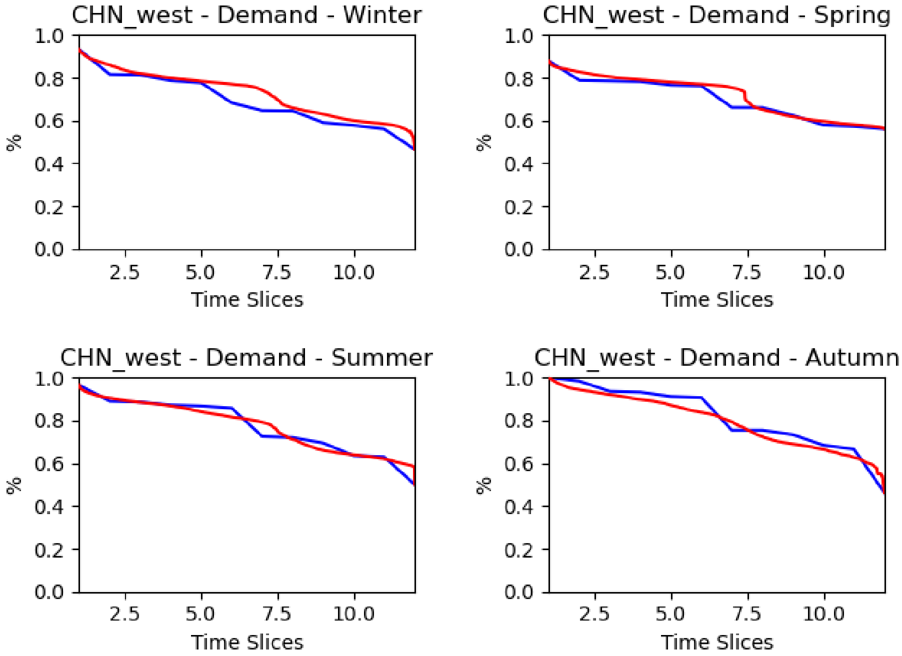

Figure 9: Seasonal demand results of 48-time slices and its comparison with original data.

4 Operational model framework

In the previous sections, we discussed the approach for selecting time slices and presented the results obtained from this approach. These results serve as input data for the direct integration of short-term variations into long-term models. In this section, we introduce a detailed operational model framework that is used to establish a soft link with the long-term model.

we consider a single wind-hydrogen system operator that is the owner of an onshore wind farm with a co-located compressed-gas-based hydrogen energy storage (HES) system. We assume that the system operator should sign a one-year contract with a steel industry at the beginning of the year. The contract stipulates the amount of electricity the operator should supply to the steel industry per hour. It is also agreed that the fluctuation of the total amount of electricity per week should be within a certain range. The goal of the system operator is to maximize the cumulative profit over time by computing the amount of electricity to commit to delivery during the time t. Figure 10 shows the framework of the model.

4.1 Indexing structure

The model uses the following indices and the corresponding sets:

- $t \in T$ discrete-time index corresponding to the decision epoch which represents the time period t.

4.1.1 Decision variables

Decision-makers control the modeled system by decision (control) variables:

- x_t the amount of electricity we commit to deliver during time t.

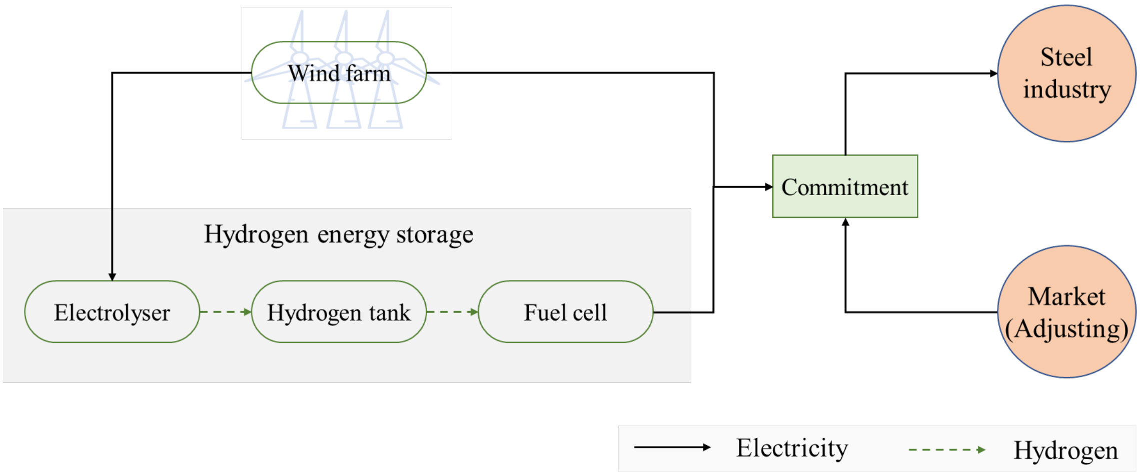

Figure 10: Model framework

• ein_t the amount of electricity used to produce hydrogen during the time t.
• $eout$ the amount of electricity generated from the fuel cell and delivered to the industry during the time t.

4.1.2 Outcome variables

Outcome variables are used for evaluation of the consequences of implementation of the decisions; therefore, at least one of them is used as the optimization objective.

In the model prototype 2 outcomes (both used as criteria in multiple-criteria model analysis) are defined:
• $profit$ the total profit of the system, and
• $penal$ the total penalty for not delivering electricity as a commitment

4.1.3 Exogenous variables

The variables define the state of the system, of which the values are given and independent change from other variables.
• y_t electricity generated from the wind farm during the time $t-1$.
• p_t electricity price in market 1 for electricity delivered as a commitment during the time t.
• $W_t = (y_t, p_t)$: Exogenous state of the system at time t.

4.1.4 State variables

The variables defining the state of the system:
• yh_t hydrogen generated from the electrolyzer during the time t.
• r_t hydrogen energy storage level in the hydrogen tank during the time t.
• $S_t = (yh_t, r_t, W_t)$ State of the system at time t.

4.1.5 Auxiliary variables

All other variables used in the SMS:

- xs_t electricity delivered from the system during the time t.
- $short_t$ electricity shortage between the system can delivery and the commitment during the time period t.
- xw_w the amount of electricity delivered during the week w (every 168 hours).

4.2 Parameters

The following model parameters are used in the model relations specified in Section 4.3:
- values of indices (members of sets) specified in Section 4.1.
- $int_s, s \in S$: initial state of storage device s.
- $efs_s, s \in S$: self discharge of storage devices s.
- $efi_s, efo_s, s \in S_h$: storage efficiency during charge and discharge.
- $efc_c, c \in C$: energy converting efficiency.
- $invs_s$: unit investment cost of storage device s.
- $invc_c$: unit investment cost of energy converting device c.
- $omcs_{es}$: unit operational cost of storage device s storing energy e.
- $omcc_{ec}$: unit operational cost of energy converting device c generating energy e.
- d_{et}: demand of energy e in the time period t.
- p_e: price of energy e in the time period t.
- q_e: penalty for shortage of energy e.
- $cap_s, s \in S$: the capacity of storage device s. $lcap_s \leq cap_s \leq ucap_s$.
- $pow_s, s \in S$: the amount of charging/discharging power required for a storage device s. $lpow_s \leq pow_s \leq upow_s$.

4.3 Relations

In this section, we list the main relations of the above model. The relations include energy balance, energy storage, penalty, benefits, and others. Details are as follows:
- Energy balance:

$$y_{t+1} = eine_t + eoutw_t \quad t \in T \tag{9}$$

$$yh_t = eth \cdot eine_t \tag{10}$$

$$hins_t = yh_t \tag{11}$$

$$houts_t = efs \cdot hins_t \tag{12}$$

$$eoutf_t = eff \cdot houts_t \tag{13}$$

- Electricity delivered from the system:

$$xs_t = (eoutw_t + eoutf_t) \tag{14}$$

- Electricity shortage:

$$short_t = x_t - xs_t \tag{15}$$

- Hydrogen storage:

$$r_{t+1} = \begin{cases} rmax \quad if \quad r_t + hins_t \geq rmax \\ r_t + hins_t \quad if \quad x_t \leq y_{t+1}, \quad r_t + hins_t < rmax \\ r_t - houts_t \quad if \quad y_{t+1} \leq x_t < xs_t \\ 0 \quad if \quad x_t \geq xs_t \end{cases} \tag{16}$$

$$r_{t+1} = r_t + hins_t - houts_t, \quad t \in T \tag{17}$$

• Ensure the hydrogen energy storage is available:

$$\frac{p_t}{q_t} < \lambda \cdot eth \cdot efs \cdot eff \tag{18}$$

• Penalty:

$$penal_t = q_t \cdot short_t \tag{19}$$

• Profit:

$$profit_t = \begin{cases} p_t \cdot x_t, \quad if \quad x_t \leq xs_t \\ p_t \cdot x_t - q_t \cdot short_t, \quad if \quad x_t > xs_t \end{cases} \tag{20}$$

$$profit_t = p_t \cdot x_t - q_t \cdot short_t \tag{21}$$

• fluctuation:

$$|xw_{w+1} - xw_w| \leq \epsilon \tag{22}$$

5 Conclusion

Renewable energy generation puts forward opportunities and challenges to the energy system. The high variability and uncertainty associated with renewable energy sources require careful consideration of energy storage to balance the supply and demand dynamics. In this report, we present the progress made in developing a model to support the long-term planning and short-term operation of renewable energy systems. The approach developed by Dr. Julian Hunt and Dr. Behnam Zakeri for dealing with and selecting aggregated data is reviewed. Results obtained from this approach are presented. Additionally, we present a symbolic model framework for the operation of a given renewable energy system. This work serves as the foundation for future analysis of harmonizing different time-period decisions in energy systems. Future work is needed to extend the model to support the planning of renewable energy system expansion and multi-product analysis involving hydrogen. This will allow for a more comprehensive analysis of the potential of renewable energy systems in meeting the energy demands of the future while taking into account the complexities and uncertainties associated with these systems.

References

[1] Hilal Bahlawan, Enzo Losi, Lucrezia Manservigi, Mirko Morini, Michele Pinelli, Pier Ruggero Spina, and Mauro Venturini. Optimization of a renewable energy plant with seasonal energy storage for the transition towards 100% renewable energy supply. *Renewable Energy*, 198:1296–1306, 2022.

[2] Torbjørn Egeland-Eriksen, Amin Hajizadeh, and Sabrina Sartori. Hydrogen-based systems for integration of renewable energy in power systems: Achievements and perspectives. *International Journal of Hydrogen Energy*, 46(63):31963–31983, 2021.

[3] Benedikt Finnah and Jochen Gönsch. Optimizing trading decisions of wind power plants with hybrid energy storage systems using backwards approximate dynamic programming. *International Journal of Production Economics*, 238:108155, 8 2021.

[4] Seán Collins, John Paul Deane, Kris Poncelet, Evangelos Panos, Robert C. Pietzcker, Erik Delarue, and Brian Pádraig Ó Gallachóir. Integrating short term variations of the power system into integrated energy system models: A methodological review. *Renewable and Sustainable Energy Reviews*, 76:839–856, 2017.

[5] Kumar Biswajit Debnath and Monjur Mourshed. Challenges and gaps for energy planning models in the developing-world context. *Nature Energy*, 3(3):172–184, 2018.

[6] Behnam Zakeri, Julian David Hunt, Murodbek Laldjebaev, Volker Krey, Adriano Vinca, Simon Parkinson, and Keywan Riahi. Role of energy storage in energy and water security in central Asia. *Journal of Energy Storage*, 50, 2022.

[7] Paul Nahmmacher, Eva Schmid, Lion Hirth, and Brigitte Knopf. Carpe diem: A novel approach to select representative days for long-term power system modeling. *Energy*, 112:430–442, 2016.