

New and Improved Methods to Characterize, Classify, and Estimate Daily Sky Conditions for Solar Energy Applications

Byung O Kang

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
In partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

In

Electrical and Computer Engineering

Kwa-Sur Tam, Chair
Scott M. Bailey
Virgilio A. Centeno
Xinwei Deng
Inyoung Kim

March 24, 2014
Blacksburg, Virginia

Keywords: Sky conditions, Synthetic generation, Solar irradiance estimation, NWS sky cover,
Solar irradiance quality, and Ramp rate

Copyright 2014, Byung O Kang

New and Improved Methods to Characterize, Classify, and Estimate Daily Sky Conditions for Solar Energy Applications

Byung O Kang

ABSTRACT

Firstly, this dissertation proposes a new characterization and classification method for daily sky conditions by using the daily sky clearness index (K_D) and the daily probability of persistence (POP- K_D) that can be derived from ground-based irradiance measurement data. Quality of daily solar irradiance is characterized by a newly proposed parameter, POP- K_D . This characterized daily quality is varying and uncertain at the middle level of the quantity, but high and more certain at very high and low quantity levels. In addition, the proposed characterization method shows interesting results for K_D and POP- K_D : a statistical consistency for multiple years and similarity for their seasonal trends. The classification results also indicate an existence of dominant classes, and transitions between the dominant classes are significant for all locations. This dissertation also generates annual synthetic sequences of K_D and POP- K_D using a Markov approach. The generated sequences show statistical similarities with observed sequences.

Secondly, this dissertation proposes methodologies to estimate day-ahead solar irradiance using the National Weather Service (NWS) sky cover forecast. For model development, this paper splits up a direct estimation process from the sky cover forecast to solar irradiance into two stages: forecast verification and cloud-to-irradiance conversion. Uncertainty for each stage and for the overall estimation process is quantified. NWS forecast uncertainty (about 20%) is identified as the main source of uncertainty for the overall process. In addition, verification of the sky cover forecast shows approximately 20% overestimated bias at days with a high

irradiance level. Thus, the NWS sky cover forecast needs to be adjusted based on the type of day. This dissertation also proposes a conversion equation relating daily quantity of cloud information and daily quantity of solar irradiance. The proposed conversion equation achieves accuracy with simplicity. Five day-ahead solar irradiance quantity estimation methods are proposed in this dissertation. The proposed methods incorporate different schemes for dealing with the bias discovered in the cloud forecast. The observed data are regularly found within the 95% confidence intervals of the estimated values. Estimation results demonstrate the effectiveness of the conditional adjustment schemes at different irradiance levels.

Lastly, this dissertation proposes a methodology to estimate day-ahead solar irradiance using fluctuation information of the NWS sky cover forecast. POP- K_D was used as a parameter for the quality of daily solar irradiance. POP- K_D efficiently represents the quality of daily solar irradiance. In addition, POP- K_D indicates the probability that solar irradiance variability is within the ramp rates of common generators in power systems at a certain photovoltaic penetration level. This dissertation also proposes a new equation for the conversion from cloud fluctuation information to daily quality of surface solar irradiance. The proposed equation achieves accuracy. The proposed day-ahead solar irradiance quality estimation method is based on fluctuation information provided by the NWS sky cover forecast. This method uses a normalization approach to relate fluctuation of cloud forecast and fluctuation of cloud observation. The observed data are regularly found within the 95% CIs of the estimated values.

DEDICATION

This dissertation is dedicated to my loving wife (Eunju Woo), my daughter (Elizabeth Jiwoo Kang), and my parents in Korea (Kyung Duck Kang and Young Sook Choi).

항상 저를 사랑으로 응원해 준 사랑스러운 아내와 딸, 그리고 한국에 계신 부모님께 이
논문을 바칩니다.

ACKNOWLEDGEMENTS

Since this dissertation was certainly a collaborative endeavor, I wish to express my sincere appreciation to many people who provided support, assistance, encouragement, and direction toward the completion of this dissertation.

First, I would like to express my deepest and most sincere appreciation to committee chairman, Dr. Kwa-Sur Tam, for his constant support, encouragement, patience, and invaluable guidance throughout my doctoral pursuit. His exceptional knowledge, research skills, wisdom, and personality have certainly inspired and enriched me in many ways, both professionally and personally. I will never be able to fully express my gratitude and respect to him for helping me to gain invaluable knowledge. I am also sincerely grateful for him to afford research opportunities to me.

Moreover, I would like to acknowledge the assistance of my other committee members, Dr. Inyoung Kim, Dr. Xinwei Deng, Dr. Virgilio A. Centeno, and Dr. Scott M. Bailey. I am grateful to Dr. Inyoung Kim and Dr. Xinwei Deng for me to conduct in-depth statistical analyses. Their advice and instruction were invaluable to the completion of this dissertation. I also wish to thank Dr. Virgilio A. Centeno and Dr. Scott M. Bailey for offering valuable insights and posing challenging questions. Their valuable suggestions and commitment significantly improved my study.

I would like to acknowledge financial support from the National Science Foundation under Grant 1048079. In addition, I owe my sincere appreciation to my colleague, Rakesh Sehgal, for help with the data acquisition. I also wish to thank my cousin in Blacksburg, Dr. Jae H. Park,

and his family who provided encouragement and help during my Ph.D. journey. I would like to extend thanks to my friends in Blacksburg for precious memories that we shared together.

Last, but not least, I am grateful to my loving wife, Dr. Eunju Woo, for providing emotional support and encouragement throughout my academic endeavors. She inspired me to begin my Ph.D. journey and offered many sacrifices during my Ph.D. life. Without her, I would never have completed my degree.

TABLE OF CONTENTS

CHAPTER 1. INTRODUCTION.....	1
1.1. Background.....	1
1.1.1. Sky conditions and surface solar irradiance	1
1.1.2. Prediction of day-ahead solar irradiance	2
1.2. Statement of the problem.....	4
1.3. Objectives of the study.....	5
1.4. Outline.....	8
 CHAPTER 2. LITERATURE REVIEW.....	 9
2.1. Characterization and classification for daily sky conditions	9
2.1.1. Characterization based on solar irradiance quantity	9
2.1.2. Characterization based on solar irradiance quality	9
2.1.3. Classification	10
2.2. Solar irradiance prediction.....	11
2.2.1. Persistence	11
2.2.2. Satellite-based cloud motion	11
2.2.3. Ground-based sky imager.....	12
2.2.4. Time series and artificial intelligence.....	12
2.2.5. NWP model	14
2.2.6. NWS sky cover forecast	15

**CHAPTER 3. NEW CHARACTERIZATION AND CLASSIFICATION OF DAILY
SKY CONDITIONS..... 17**

3.1. Chapter introduction	17
3.2. Data	18
3.3. Characterization and classification	19
3.3.1. New characterization method	19
3.3.2. New classification method	26
3.4. Results.....	29
3.4.1. Consistent patterns	29
3.4.2. Identification of dominant sky conditions	36
3.4.3. Monthly patterns of dominant classes	39
3.4.4. Seasonal patterns of K_D and POP- K_D	41
3.5. Generation of K_D and POP- K_D sequences	50
3.6. Chapter contributions.....	60

**CHAPTER 4. DAY-AHEAD QUANTITY ESTIMATION OF SOLAR IRRADIANCE
USING SKY COVER FORECAST 62**

4.1. Chapter introduction	62
4.2. Data	63
4.3. Relationship between observed and forecast sky cover.....	64
4.3.1. Ground-based sky cover observation	64

4.3.2. NWS day-ahead total sky cover	65
4.3.3. Relationship between $TKC_{D,NWS}$ and TKC_D	68
4.3.4. Verification	71
4.4. Relationship between observed sky cover and ground-based solar irradiance.....	73
4.4.1. Relationship between TKC_D and K_D	73
4.4.2. Verification	77
4.5. Proposed estimation methods for day-ahead K_D	79
4.5.1. Proposed Method 1.....	79
4.5.2. Proposed Method 2.....	83
4.5.3. Proposed Method 3.....	84
4.5.4. Proposed Method 4.....	85
4.5.5. Proposed Method 5.....	86
4.6. Results.....	89
4.7. Chapter contributions.....	96
CHAPTER 5. DAY-AHEAD QUALITY ESTIMATION OF SOLAR IRRADIANCE USING SKY COVER FORECAST	97
5.1. Chapter introduction	97
5.2. Data	98
5.3. Solar irradiance quality	99
5.3.1. Meaning of POP- K_D	99
5.3.2. Statistics of POP- K_D	100
5.3.3. Ramp rate	109

5.4. Relationship between quality of observed sky cover and quality of forecast sky cover	113
5.4.1. Relationship between POP-TKC _{D,NWS} and POP-TKC _D	113
5.4.2. Verification	115
5.5. Relationship between quality of observed sky cover and quality of solar irradiance.....	117
5.5.1. Relationship between POP-TKC _D and POP-K _D	117
5.5.2. Verification	117
5.6. Proposed estimation method for day-ahead POP-K _D	120
5.7. Results.....	123
5.8. Chapter contributions.....	126
CHAPTER 6. CONCLUSION AND FUTURE WORK	127
6.1. Conclusion	127
6.2. Contribution	131
6.3. Future works	133
APPENDIX A. MATLAB[®] SOURCE CODE	134
A.1. Description of MAT-files.....	134
A.1.1. Golden_2001_2011_YearlyGroup_idx_perK_proS.mat	134
A.1.2. UNLV_2007_2011_YearlyGroup_idx_perK_proS.mat.....	134
A.1.3. ORNL_2008_2011_YearlyGroup_idx_perK_proS.mat.....	135
A.1.4. AugiarMTM_All_Locations_Monthly_perK_proS_classes.mat.....	135
A.1.5. Golden_2009_2012_YearlyGroup_ClassGroup_TKCDfor_TKCDobs.mat	135

A.1.6. Golden_2001_2012_YearlyGroup_KD_POPKD_Class_stdQD_kurto.mat	136
A.1.7. Golden_2009_2012_YearlyGroup_POPTKCDfor_POPTKCDobs.mat	136
A.1.8. Golden_2001_2011_YearlyGroup_MonthlyGroup_GHI.mat.....	137
A.1.9. Golden_2006_2012_YearlyGroup_TKCD_OKCD_POPTKCD_POPOKCD.mat.....	137
A.1.10. Golden_2012_YearlyGroup_KD_POPKD_TKCD_POPTKCD.mat.....	138
A.2. M-files for Chapter 3.....	139
A.2.1. Annual characterization results at SRRL BMS, UNLV, and ORNL (Section 3.4.1)	139
A.2.2. Annual means of K_D and POP- K_D at SRRL BMS, UNLV, and ORNL (Section 3.4.1).....	154
A.2.3. MTMs and monthly patterns at SRRL BMS, UNLV, and ORNL (Section 3.4.2 & 3.4.3)	156
A.2.4. 3-dimensional CDFs for K_D and POP- K_D at SRRL BMS, UNLV, and ORNL (Section 3.4.4)	159
A.2.5. Generation of synthetic sequences (Section 3.5)	164
A.3. M-files for Chapter 4.....	172
A.3.1. Bias between $TKC_{D,NWS}$ and TKC_D (Section 4.3)	172
A.3.2. Curve fitting between TKC_D and K_D (Section 4.4)	172
A.3.3. Description of the proposed Methods 1 – 5 (Section 4.5)	174
A.3.4. Tests for the proposed Methods 1 – 5 (Section 4.6)	176
A.4. M-files for Chapter 5.....	207
A.4.1. 3-D synoptic view of K_D , POP- K_D , and standard deviation/kurtosis (Section 5.3.2)	207
A.4.2. Ramp rate of I_{GHI} for 11 years (Section 5.3.3).....	209
A.4.3. Relationship between POP- $TKC_{D,NWS}$ and POP- TKC_D for 3 years (Section 5.4)	215
A.4.4. Relationship between POP- TKC_D and POP- K_D for 3 years (Section 5.5).....	218
A.4.5. Description of the proposed quality method (Section 5.6)	221
A.4.6. Tests for the proposed quality method (Section 5.7)	222
REFERENCE.....	227

LIST OF FIGURES

Fig. 3.1: Process and example for calculating POP- K_D	20
Fig. 3.2: The 10 classes defined by the proposed characterization and classification methods. ..	22
Fig. 3.3: Examples of the daily solar irradiance plot of each class from sunrise to sunset.	28
Fig. 3.4: Annual and multi-year plots for SRRL BMS (2001 ~ 2011).	31
Fig. 3.5: Annual and multi-year plots for UNLV (2007 ~ 2011).	32
Fig. 3.6: Annual and multi-year plots for ORNL (2008 ~ 2011).....	33
Fig. 3.7: Ranges (annual mean \pm 1 standard deviation) for (a) K_D and (b) POP- K_D for 11 years at SRRL BMS, 5 years at UNLV, and 4 years at ORNL.....	35
Fig. 3.8: Monthly patterns of Class 1, Class 4, and Class 5 in UNLV.	40
Fig. 3.9: Monthly patterns of Class 1, Class 4, Class 5, and Class 8 in SRRL BMS.	40
Fig. 3.10: Monthly patterns of Class 1, Class 4, Class 5, Class 6, and Class 8 in ORNL.	41
Fig. 3.11: Seasonal cumulative frequency distributions of (a) K_D and POP- K_D , (b) K_D , and (c) POP- K_D and (d) seasonal relationship between K_D and POP- K_D in SRRL BMS.	45
Fig. 3.12: Seasonal cumulative frequency distributions of (a) K_D and POP- K_D , (b) K_D , and (c) POP- K_D and (d) seasonal relationship between K_D and POP- K_D in UNLV.....	47
Fig. 3.13: Seasonal cumulative frequency distributions of (a) K_D and POP- K_D , (b) K_D , and (c) POP- K_D and (d) seasonal relationship between K_D and POP- K_D in ORNL.....	49
Fig. 3.14: Synthetic and observed sequences of K_D in 2011 for (a) SRRL BMS, (b) UNLV, and (c) ORNL.	56
Fig. 3.15: Synthetic and observed sequences of POP- K_D in 2011 for (a) SRRL BMS, (b) UNLV, and (c) ORNL.	58

Fig. 3.16: K-S statistic observation of synthetic and observed sequences at three locations for (a) K_D and (b) POP- K_D	59
Fig. 4.1: Retrieval and processing of the NWS total sky cover; (a) retrieval of historical data from NDFD and (b) example of how the total sky cover forecasts are processed.	66
Fig. 4.2: Distribution of $TKC_{D,bias}$ for (a) overall 3-year observation and for days with (b) high K_D , (c) medium K_D , and (d) low K_D levels.....	70
Fig. 4.3: Relationship between observed TKC_D and measured K_D	74
Fig. 4.4: Relationship between observed TKC_D and Box-Cox transformed K_D	75
Fig. 4.5: Algorithms for the proposed Method 1.	80
Fig. 4.6: Conversion curves and the boundaries for the 95% confidence interval (CI) for the proposed Methods 1, 2, 3, and 4.	82
Fig. 4.7: Algorithms for the proposed Method 2.	83
Fig. 4.8: Algorithms for the proposed Methods 3 and 4.....	84
Fig. 4.9: Algorithms for the proposed Method 5.	85
Fig. 4.10: Gap generated by adjustment in Stage 2A.	87
Fig. 4.11: Conversion curve and the 95% CI boundaries for Method 5.	88
Fig. 4.12: K–S statistics for the five proposed quantity estimation methods: (a) CDFs and (b) differences of CDFs generated by the proposed methods and the CDF of the observed data.....	91
Fig. 4.13: Comparison of the estimated and observed K_D with the 95% CIs for the first 31 days of the year 2012 using (a) Method 1, (b) Method 2, (c) Method 3, (d) Method 4, and (e) Method 5, respectively.....	95
Fig. 5.1: Examples of how the per-minute k remains within the POP- K_D threshold: (a) relatively lower fluctuation and (b) relatively higher fluctuation.	100

Fig. 5.2: Statistical relationship between K_D and POP- K_D for 11 years (2001 – 2011) at SRRL BMS; (a) cumulative data points and (b) boxplots grouped by different K_D	102
Fig. 5.3: Examples of daily probability density function (PDF) of $\Delta\tilde{k}$ at SRRL BMS in case of (a) high K_D , (b) low K_D , (c) medium K_D with high POP- K_D , (d) medium K_D with medium POP- K_D , and (e) medium K_D with low POP- K_D	105
Fig. 5.4: 3-D synoptic view of K_D , POP- K_D , and standard deviation (σ) at SRRL BMS for 11 years (2001 – 2011).	107
Fig. 5.5: 3-D synoptic view of K_D , POP- K_D , and kurtosis at SRRL BMS for 11 years (2001 – 2011).	108
Fig. 5.6: CDFs of ramp rates of I_{GHI} for overall 11-year (2001 – 2011) observation at SRRL BMS and for days with high K_D , medium K_D , and low K_D levels: (a) entire CDFs and (b) graphs zoomed between 0 and 150W/m^2 for x-axis and between 0.8 and 1 for y-axis.....	111
Fig. 5.7: Comparison of CDFs of POP-TKC $_D$ and POP-TKC $_{D,NWS}$ for 3-year (2009 – 2011): (a) original values and (b) normalized values.	114
Fig. 5.8: Relationship between POP-TKC $_D$ and POP- K_D at SRRL BMS for 3 years (2009 – 2011).	116
Fig. 5.9: Algorithm for the proposed quality estimation method.	118
Fig. 5.10: Overall sequence of the proposed estimation methods for quantity and quality of daily solar irradiance.....	119
Fig. 5.11: Conversion lines and the boundaries for the 95% confidence interval (CI) for the proposed quality estimation method.	122

Fig. 5.12: K–S statistics for the proposed quality estimation method: (a) CDFs and (b) differences of the CDF generated by the proposed method and the CDF of the observed data.

..... 125

Fig. 5.13: Comparison of the estimated and observed POP- K_D with the 95% confidence intervals (CIs) for the first 31 days of the year 2012. 125

LIST OF TABLES

Table 3.1: Information about the three locations where the solar data in this study are obtained.	18
Table 3.2: Definition and description of the proposed 10 classes.	23
Table 3.3: Occurrence percentages and average values of K_D ($K_{D,ave}$) and POP- K_D (POP- $K_{D,ave}$) of the proposed 10 classes at three locations.	25
Table 3.4: Range of POP- K_D for different values of K_D on the irregular nonagon boundary.	31
Table 3.5: Transition matrix and frequency distribution of classes in SRRL BMS.	37
Table 3.6: Transition matrix and frequency distribution of classes in UNLV.	37
Table 3.7: Transition matrix and frequency distribution of classes in ORNL.	38
Table 3.8: Comparison of seasonal trends for K_D (part b), POP- K_D (part c), and K_D & POP- K_D (part d) in Fig. 3.11, 3.12, and 3.13.	43
Table 3.9: Maximum, minimum, and number of months of K_D and POP- K_D in 10 classes for generating a library of MTMs.	51
Table 3.10: KSI and OVER results of synthetic and observed sequences of K_D and POP- K_D in Fig. 3.14 and 3.15.	53
Table 3.11: Annual averages of synthetic and observed sequences of K_D and POP- K_D in Fig. 3.14 and 3.15.	54
Table 3.12: Occurrence percentages of the proposed classes based on observed multi-year data and average of 100 generated synthetic sequences at three locations.	54
Table 4.1: A sample of the absolute differences between the next-day total sky cover forecast released by the National Weather Services at 18:00 and at midnight for SRRL BMS.	67

Table 4.2: Mean and standard deviation of $TKC_{D,bias}$ at different K_D levels for three years (2009 – 2011).	70
Table 4.3: Mean and standard deviation of ΔTKC_D at different K_D levels for one year (2012). .	72
Table 4.4: Mean and standard deviation of ΔK_D at different K_D levels for one year (2012).	77
Table 4.5: Comparison of RMSEs for the proposed quantity estimation methods.	89
Table 4.6: Comparison of MBEs for the proposed quantity estimation methods.....	90
Table 4.7: Comparison of KSI and OVER values for the proposed quantity estimation methods.	90
Table 4.8: Percentage of days that the observed K_D is within the envelope of the proposed quantity estimation methods at different K_D levels for one year overall (2012).	93
Table 5.1: Boxplot result of POP- K_D grouped by different K_D ranges.....	103
Table 5.2: K_D , POP- K_D , standard deviation, and kurtosis of the examples in Fig. 5.3.	106
Table 5.3: RMSE and MBE of $\Delta POP-TKC_D$ for 1 year (2012).	116
Table 5.4: RMSE and MBE of $\Delta POP-K_D$ for 1 year (2012).	118
Table 5.5: Comparison of RMSE and MBE of the proposed quality method for one year (2012)	124
Table 5.6: Percentage of days that the observed POP- K_D is within the CIs of the proposed quality method (hit rate) for one year (2012).....	124

CHAPTER 1. INTRODUCTION

1.1. Background

1.1.1. Sky conditions and surface solar irradiance

Solar irradiance received at ground level at a site is very much influenced by the sky conditions because extraterrestrial solar irradiance is attenuated by three atmospheric processes: scattering, absorption, and reflection. As the solar-based industry continues to grow, more sophisticated characterization of sky condition is required for better utilization of solar irradiance. For several decades, a sky clearness index has been used to characterize the quantity of solar irradiance since the pioneering work by Liu and Jordan (1960). The sky clearness index is the ratio of actual obtained global solar irradiance at the surface to extraterrestrial irradiance (Iqbal, 1983). More details about this index are mentioned in Chapter 3.

The sky clearness index can be a good indicator when conducting quantitative observations over a given time period. Sky conditions can be distinguished from bimodality on the distribution of the short-term clearness index. The bimodality corresponds to clear and cloudy conditions. The bimodal behavior is also shown at a daily clearness index, the ratio of integrated global solar irradiance at the surface to integrated extraterrestrial irradiance during a day. However, the bimodality of the daily observation is less significant than that of the short-term observation. Thus, the less bimodal characteristic indicates the existence of intermediate states between clear and cloudy conditions.

The quality of solar irradiance during a period can be considered to be fluctuation of the short-term solar irradiance at ground level. The short-term solar irradiance at ground level fluctuates due to changes in the atmosphere such as cloud formation and cloud movement. Analysis of the solar irradiance fluctuation is also valuable for the characterization of sky conditions. The fluctuation of solar irradiance may damage power electric devices or grid-connect residents in a solar-based power generation system. In addition, the fluctuation can be a considerable factor when deciding power storage size of back-up facilities in remote areas (Woyte et al., 2007). Therefore, if solar irradiance is of high quality, the need for the grid or storage facility to smoothen the fluctuation is reduced.

1.1.2. Prediction of day-ahead solar irradiance

As increasing amounts of solar power are being used to satisfy electric demands, solar irradiance prediction becomes important for reliable power operations. Due to the variability of solar resources, the solar industry needs more accurate prediction results. Accurate prediction results will enable power operators to make better scheduling decisions for solar-based power generation and to avoid excessive backup facilities.

The prediction time scales for the operation of the power grid are classified by minutes-ahead, hours-ahead, and days-ahead. The minutes- and hours-ahead scales are sometimes called intra-day prediction. According to Heinemann et al. (2006), different application areas are favored by different prediction time scales. Firstly, the minutes-ahead prediction results might be useful for the Solar Thermal Power Plant (STPP). Secondly, the hour-ahead prediction results may apply to building controls. Lastly, the days-ahead prediction results can be more favored for Photovoltaic (PV) systems.

The solar irradiance prediction can be divided by cloud image-based approaches and Numerical Weather Prediction (NWP) model-based approaches. Analysis of cloud images is useful for solar irradiance prediction because the ground-based solar irradiance is directly affected by the extent of cloud coverage. Conventionally, cloud images are rendered from satellite-based remote sensing. In recent years, researchers have increasingly focused on developing automated image-processing schemes using ground-based sky imagers because they can provide more frequent cloud information with higher resolution (Pfister et al., 2003; Long et al., 2006; Cazorla et al., 2008). However, stand-alone use of sky imager-based images for day-ahead prediction is not feasible since cloud images cannot be collected and analyzed at night. Thus, the sky imager-based prediction method is only favored for use in intra-day prediction.

The NWP model-based approach overcomes the temporal limitations of the sky imager-based approach by extending the prediction time horizon several days ahead. In addition, the Model Output Statistics (MOS) technique with the NWP models can reduce the prediction errors. Further details about the NWP models and the MOS technique are described in Chapter 2.

A prediction method using sky cover forecasts released by the National Weather Service (NWS) originates from the NWP-MOS approach. The NWS provides free and publicly available gridded forecasts of sky cover, the forecasted percentage of cloud coverage in the atmosphere. The NWS sky cover forecast combines inputs from human experts and automated measurements from NWS offices at various locations; the National Centers for Environmental Prediction (NCEP) provides simulation results using NWP models and satellite-based information (Perez et al., 2007; Perez et al., 2010; Marquez and Coimbra, 2011). Thus, the NWS sky cover forecast provides a fairly accurate day-ahead prediction of cloud coverage, and this forecast cloud data can be converted into the solar irradiance based on an empirical fit (Perez et al., 2007). This

approach has been in operation and favored for day-ahead solar irradiance prediction (Perez et al., 2007; Remund et al., 2008; Perez et al., 2010; Kleissl, 2010; Perez et al., 2013). Further specific reviews about the NWS sky cover forecast-based approaches are described in Chapter 2.

1.2. Statement of the problem

Previous researches have shown that the characterization for sky conditions based on the short-term clearness index, which has a time resolution of less than five minutes, tends to have a bimodal characteristic (Suehrche and McCormick, 1988; Skartveit and Olseth, 1992; Jurado et al., 1995; Tovar et al., 1998). The bimodality can correspond to clear and cloudy conditions. Daily observation for the sky conditions using a daily clearness index also shows the bimodal behavior, but it is less significant than that of the short-term (Skartveit and Olseth, 1992; Hansen, 1999; Ibanez et al., 2002, 2003). Fewer bimodal characteristics on daily observation indicate the existence of intermediate states between clear and cloudy conditions.

To understand the intermediate states between clear and cloudy conditions, most previous researches focused on quantitative analysis for daily solar irradiance and relatively limited researches focused on qualitative analysis. In addition, the quantitative and qualitative analyses were separately conducted in previous approaches. Thus, the current research examines the intermediate states based on a simultaneous analysis of both daily solar irradiance quantity and quality. For this analysis, the parameter for the solar irradiance quantity and the parameter for the solar irradiance quality must be compatible. While the sky clearness index has been commonly utilized for quantity in the previous studies, different parameters have been applied for the fluctuation of solar irradiance. Therefore, the current research proposes a new parameter for the

quality, which is able to be compatible with the quantity parameter and efficiently represents daily solar irradiance quality.

Prediction of day-ahead sky conditions is directly related to predicting day-ahead values of the quantity and quality parameters. As previously mentioned, the prediction method using the NWS sky cover forecast has been in operation for day-ahead solar irradiance prediction and its performance has been validated by comparisons with other prediction models in previous studies (Perez et al., 2007; Remund et al., 2008; Perez et al., 2010; Perez et al., 2013). This approach used in the previous studies directly converted the forecast cloud into surface solar irradiance. However, prediction accuracy can be improved by uncertainty investigation on the overall conversion process. In addition, advanced processes need to obtain accurate daily cloud quantity and quality from the forecast products for day-ahead prediction. Therefore, the current research has developed new and improved methods to predict day-ahead values of the quantity and quality parameters based on modifications of the previously introduced NWS sky cover forecast-based approach.

1.3. Objectives of the study

The first primary goal of this dissertation is to characterize and classify daily sky conditions with quantitative and qualitative analyses based on ground-based solar irradiance measurement data. A joint use of quantity and quality parameters for daily solar irradiance can facilitate daily sky conditions to be more specified and diversified. Thus, this dissertation will propose a new parameter for the qualitative analysis of daily solar irradiance which can be

compatibly used with the quantity parameter. Using this newly proposed quality parameter, new characterization and classification methods will be developed.

The second primary goal of this dissertation is to improve and develop methodologies for accurately estimating day-ahead sky conditions using the NWS sky cover forecast based on the proposed characterization and classification results. Various cases of daily solar irradiance quality exist under a certain case of daily solar irradiance quantity. The sky condition is correlated to this relationship between the quantity and quality of solar irradiance. This indicates that the quantity estimation and the quality estimation are independent. Thus, the estimation methods for quantity and quality will be separately developed in this research.

The specific objectives for this research are summarized as follows.

- This research attempts to propose a new parameter for the qualitative analysis of daily solar irradiance. The proposed parameter will be evaluated to determine how the newly proposed quality parameter efficiently represents the daily solar irradiance quality. In addition, this research attempts to apply the proposed quality parameter for a solar-based generation system.
- Based on the proposed quality parameter, this research develops a new method to characterize daily sky conditions including quantitative and qualitative analyses for daily solar irradiance.
- Based on the proposed characterization method, this research develops a new method to classify daily sky conditions into an increased number of specific classes.
- The newly proposed characterization and classification methods will be tested by multi-year solar irradiance measurement data at different locations in the continental United States.

Thus, this research attempts to find interesting relationships between daily solar irradiance quantity and quality from the test results.

- This research attempts to generate the synthetic sequences of the quantity and quality parameters based on a stochastic approach. These generated synthetic sequences may be useful for substituting the limited observed data at a selected location.
- This research attempts to improve the previously introduced conversion from the NWS sky cover forecast to surface solar irradiance quantity based on uncertainty quantification for the overall conversion process.
- This research attempts to observe the relationship between the daily quantity of cloud observation and the daily quantity of surface solar irradiance.
- Based on the uncertainty investigation and observed relationship, this research develops improved methods for estimating day-ahead solar irradiance quantity based on daily quantity of the NWS sky cover forecast.
- This research applies the proposed quality parameter for analyzing the daily fluctuation of the observed cloud.
- This research attempts to newly define the relationship between daily cloud fluctuation and daily fluctuation of surface solar irradiance.
- Based on the analyzed cloud fluctuation and the defined relationship, this research develops a new method for estimating day-ahead solar irradiance quality based on the daily quality of the NWS sky cover forecast.
- The proposed quantity and quality estimation methods will be tested by various statistical parameters.

1.4. Outline

This dissertation consists of six chapters including this introduction chapter. Chapter 2 presents a literature review of previously introduced characterization and classification methods for daily sky conditions and solar irradiance prediction methods based on observed and forecast sky cover. Chapter 3 proposes new characterization and classification methods for daily sky conditions based on ground-based solar irradiance measurement data. Chapter 4 proposes new five day-ahead solar irradiance quantity estimation methods based on the daily quantity of sky cover forecast. Chapter 5 proposes a new day-ahead solar irradiance quality estimation method based on daily fluctuation information of sky cover forecasts. The conclusion and future works are presented in Chapter 6.

CHAPTER 2. LITERATURE REVIEW

2.1. Characterization and classification for daily sky conditions

2.1.1. Characterization based on solar irradiance quantity

The characterization of sky condition was categorized based on the clearness index in the literature (Zangvil and Aviv, 1988; Zangvil and Lamb, 1997; Udo, 2000; Assuncao et al., 2007; Che et al., 2007). Zangvil and Aviv (1988) characterized sky condition into 5 categories based on a relationship between the clearness index and diffuse/global irradiance in Sde Boker, Israel. Zangvil and Lamb (1997) extended Zangvil and Aviv (1988) by removing zenith angle influences on the relationship. Characterization using the clearness index and relative sunshine was conducted in Ilorin, Nigeria (Udo, 2000). Che et al. (2007) characterized sky condition based on correlations of the index-relative sunshine and diffuse ratio-relative sunshine in 14 stations in China. Using frequency distributions of clearness index and relative optical air mass, Assuncao et al. (2007) characterized sky condition as four states in Botucatu, Brazil.

2.1.2. Characterization based on solar irradiance quality

Different studies show the characterization of sky condition based on solar irradiance quality using different methods (Suehrche and McCormick, 1988; Otani et al., 1997; Kitamura, 1999; Kawasaki et al., 2006; Woyte et al., 2007). Suehrche and McCormick (1988) used an investigation into correlations of the short-term clearness indices for the fluctuation analysis with the fast transition. Otani et al. (1997) employed a statistical method for the fluctuation analysis with the fluctuation factor: the root mean square of signal processed fluctuation of time series

and the power spectral density of the time series. Kitamura (1999) proposed the three parameters: the ratio of magnitude, duration, and rate of a transition between cloudy and clear sky for describing power fluctuations. Kawasaki et al. (2006) combined the Fourier transform and the wavelet transform based on the time series of solar irradiance. Woyte et al. (2007) demonstrated wavelet-based spectral analysis using time series of the short-term clearness index with 1- and 5-second sampling period.

2.1.3. Classification

For daily classification based on the ground-based solar irradiance, three classes are typical: clear, partly overcast, and entirely overcast sky. Maafi and Harrouni (2003) used daily fractal and clearness indices for the classification, and Harrouni et al. (2005) continued this approach using the 10-minute based solar data. Similarly, Rahim et al. (2004) and Baharuddin et al. (2010) conducted based on cloud ratio and sunshine duration. The higher number of classification was also introduced. Soubdhan et al. (2009) classified four distinct daily distributions of short-term (1 second) clearness index based on a mixture of Dirichlet distributions. Calbo et al. (2001) classified into five sky conditions based on cloud observation with clearness indices, diffuse fraction, and statistical parameters of global solar irradiance. Seven sky conditions were introduced using illuminance and solar irradiance data (Umemiy and Kanou, 2008).

2.2. Solar irradiance prediction

2.2.1. Persistence

The persistence prediction method assumes current solar irradiance will persist and that close future solar irradiance will be identical with the present solar irradiance. This method is significantly simple and deterministic. The performance of this method highly depends on the frequency of cloud fluctuation. The prediction accuracy significantly decreases when cloud rapidly changes. Thus, this method is frequently used as a benchmark (Heinemann et al, 2006).

2.2.2. Satellite-based cloud motion

The satellite-based cloud motion prediction predicts solar irradiance using consecutive satellite-derived images to track pixel-specific cloud motion that includes information on motion, formation, and dissolution of clouds (Hammer et al., 1999). According to Heinemann et al. (2006), the satellite-based cloud motion prediction reduces the errors compared to both the satellite-to-irradiance nowcasting model and the persistence prediction model. The satellite-based cloud motion prediction is considered as the best prediction technique for the 1- to 5-hour prediction time range. In addition, it has a high spatial availability because the satellite-derived imagery can cover the places where the ground-based measurements do not exist. However, it can be only used for the intra-day prediction because the visible channel images from sunrise and sunset of the next day cannot be accessible.

2.2.3. Ground-based sky imager

Solar irradiance prediction using the ground-based sky imager is favored for intra-day prediction (Casa Nova et al., 2005; Crispim et al., 2008; Chow et al., 2011; Marquez et al., 2013; Marquez and Coimbra, 2013). Casa Nova et al. (2005) proposed the combination of two methods: prediction of atmosphere attenuation based on the ground-measured solar irradiance and prediction of cloud evolution based on the sky imager-processed cloud images. Crispim et al. (2008) used the sky imager for extracting sky cover observation using a specific pixel classifier and applied these results into an Artificial Neural Network (ANN)-based prediction model. Chow et al. (2011) employed sky imager-based cloud images to determine sky cover using a library of clear sky and sunshine parameter. Their results indicate that the sky imager-based approach is beneficial to 15 – 20 minutes prediction horizons. Marquez et al. (2013) used cloud information obtained from a sky imager and Infrared Radiometric (IR) measurements for the ANN-based prediction model. Their work showed the improved prediction results for 1-hour forecast of global horizontal irradiance. Marquez and Coimbra. (2013) introduced a methodology to forecast direct normal irradiance for 1-minute forecast horizons using data obtained from the sky imager.

2.2.4. Time series and artificial intelligence

Time series statistical methods have been used to find the stationary and sequential characteristics of solar irradiance. These methods use either solar irradiance values or the sky clearness index values. The precise decision of parameters and coefficients is required since the relationship between different coefficients corresponds to different series (Mora-Lopez and

Sidrach-de-Cardona, 1998). The time series techniques predict solar irradiance hour by hour using hourly measurement data or day by day using daily averaged data. Reikard (2009) introduced the simple time-series prediction method based on regression in a log scale with time-varying coefficients. Harvey (1990) introduced an unobserved components model, which adds sinusoidal sequences into the regression method. This method was also expressed into the state equation, and the Kalman filter was applied to find the maximum likelihood function for the state representation. Reikard (2009) also used the autoregressive integrated moving average (ARIMA) method. This study specified the ARIMA orders for the hour-ahead solar irradiance prediction and used various causal inputs: extraterrestrial components, humidity, turbulence coefficients, the Richardson number, and opaque/overall cloud cover.

Artificial intelligence is another popular statistical technique for the solar irradiance prediction. These approaches frequently use various metrological parameters (air temperature, relative humidity, wind speed, wind direction, cloud index, sunshine duration, the sky clearness index, pressure, and others) and geographical coordinates (latitude, longitude, and altitude). Some prediction methods were developed based on Artificial Neural Network (ANN) (Al-Alawi and Al-Hinai, 1998; Mohandes et al., 1998; Dorvlo et al., 2002; Sözen et al., 2004; Alam et al., 2006; Mellit and Kalogirou, 2008). The Fuzzy Logic (FL)-based approaches were also introduced (Şen, 1998; Gautam and Kaushika, 2002; Gomez and Casanovas, 2003; Tulcan-Paulescu and Paulescu, 2008). Mellit et al. (2007) incorporated the FL-based approach into the ANN-based approach in order to improve performance of ANN. Mellit et al. (2006) proposed the combination of the wavelet approach and feed-forward ANN. This method shows effectiveness on solving the poor convergence issues appeared in conventional stand-alone ANN models. Paoli et al. (2009) introduced the ARIMA-ANN cascade method. The ARIMA was used to analyze

only the linear structure and the ANN was developed to capture the residuals from the ARIMA, which contain nonlinearity information.

In general, the time series, the artificial intelligence, and the integration of the time series and the artificial intelligence show acceptable performance on very short-term prediction, but high error rate on hour-ahead or day-ahead prediction (Reikard, 2009).

2.2.5. NWP model

The NWP model-based solar irradiance prediction approach attempts to mathematically solve physical energy equations relating to motion of air and water vapor in atmosphere (Kratzenberg et al., 2008; Mathiesen and Kleissl, 2011). While the previously aforementioned prediction methods show better performance for very short-term solar irradiance prediction, the NWP models are considered to be most appropriate for day- and days-ahead prediction time horizons (Perez et al., 2013). Different NWP models in different countries worldwide are currently in operation. The NWP models can be simply categorized into global and regional models based on spatial resolution scales. A typical global model in the US is the Global Forecast System (GFS) from the National Oceanic and Atmospheric Administration (NOAA) at NCEP (NCEP, 2013). This model has the resolution of 28km for the first 8 days and 35km for the next 8 days (total 16 days). Typical US regional NWP models include North American Model (NAM) and Rapid Update Cycle (RUC) (NCDC, 2013).

Statistical correction methods are applied to improve accuracy of the NWP models. Model Output Statistics (MOS) is the well-known technique. The NWP models show credible prediction results on upper air patterns, but local variations in surface tend to degrade the prediction performance of the models. Thus, the MOS uses statistical correlations between local

meteorological measurements and modeled parameters to minimize localized errors the prediction process (Mathiesen and Kleissl, 2011; Perez et al., 2013).

Some literature compared solar irradiance prediction performances using different NWP models for the continental U.S. (Remund et al., 2008; Mathiesen and Kleissl, 2011; Perez et al., 2013). Remund et al. (2008) compared three different models at three locations (Desert Rock, Boulder, and Goodwin Creek) in the U.S. continental. The percentage of the root mean squared error for day-ahead prediction is varying from 18 – 50% depending on the models and locations. Mathiesen and Kleissl (2011) validated the intra-day prediction performance using NAM, GFS, and European Centre for Medium-Range Weather Forecasts (ECMWF). Perez et al. (2013) used ten different operational models to test days-ahead prediction for the U.S. as well as Canada and Europe. The average of models' prediction output shows the better performance than individual models' prediction output.

2.2.6. NWS sky cover forecast

As one of the NWP-based prediction models, solar irradiance prediction based on a conversion from the NWS sky cover forecast to solar irradiance has been in operation for day-ahead solar irradiance prediction (Kleissl, 2010). Since the forecast product is derived from the National Digital Forecast Database (NDFD), this approach is called as NDFD. First, the NWS sky cover forecast is predicted by NOAA's GFS model, and then regional NWS offices modify the model output using the MOS technique (Perez et al., 2010).

A pioneering study (Perez et al., 2007) firstly introduced this approach as a function of the NWS sky cover forecast with the 3-hour forecast horizon at a specific location. Remund et al.(2008) tested this approach as well as other NWP models in the continental U.S. Perez et al.

(2010) modified this conversion by adjusted fitting results with the hourly interpolated time horizon. This study tested the modified conversion at several U.S. locations for both intra-day performance and days-ahead performance. Variations of this approach are also under operation (Perez et al., 2013).

CHAPTER 3. NEW CHARACTERIZATION AND CLASSIFICATION OF DAILY SKY CONDITIONS

3.1. Chapter introduction

This chapter proposes a new characterization method, K-POP method, capable of both quantitative and qualitative analyses using the ground-based irradiance measurement data. The K-POP method characterizes daily sky condition using K_D and POP- K_D . POP- K_D is a newly introduced measure for the quality of daily solar irradiance received at ground level, and the high POP- K_D indicates more steady irradiance capable of providing higher quality electric power. Further details about POP- K_D are described in Section 3.3. This chapter also proposes a new classification method cutting symmetrically the area plotted by the K-POP method into ten rectangular regions (classes). These ten classes provide ten specified quantity and quality information of daily sky conditions. The both methods are developed by using ground-based irradiance measurement data collected at three U.S. stations: Radiation Research Laboratory Baseline Measurement System (SRRL BMS), University of Nevada – Las Vegas (UNLV), and Oak Ridge National Laboratory (ORNL).

In this chapter Section 3.2 presents the details about solar irradiance data and solar resource information in the selected stations. Section 3.3 describes the measures for quantity (K_D) and quality (POP- K_D). The new characterization and classification methods are also proposed here. In Section 3.4, this chapter shows the characterization and classification results applying to multi-year data in the three stations. The local monthly and seasonal observation results are also

compared here. Section 3.5 describes synthetic generation of K_D and POP- K_D sequences. Section 3.6 provides contributions of this chapter.

3.2. Data

Solar irradiance data were obtained from Measurement and Instrumentation Data Center (MIDC) in National Renewable Energy Laboratory (NREL) (MIDC, 2012). Among the centers across the U.S., this study selected three stations: SRRL BMS, UNLV, and ORNL. Table 3.1 includes spatial and temporal information about the three locations. The three locations have different potentials for solar-based power generation due to their different local climatic conditions. The selected solar data are global horizontal irradiance measured in each station with one minute time resolution, shortest time duration of the data. All data are filtered out from sunrise to sunset every day to improve observation accuracy.

Table 3.1: Information about the three locations where the solar data in this study are obtained.

Location	Latitude (°N)	Longitude (°W)	Altitude (m)	Period	Data Time Resolution
SRRL BMS (Golden, CO)	39.74	105.18	1829	2001 ~ 2011	1-minute
UNLV (Las Vegas, NV)	36.06	115.08	615	2007 ~ 2011	1-minute
ORNL (Oak Ridge, TN)	35.93	84.31	245	2008 ~ 2011	1-minute

3.3. Characterization and classification

3.3.1. New characterization method

The new characterization method (K-POP method) is proposed based on K_D and POP- K_D . First, K_D is the daily sky clearness index. Sky clearness index indicates a relation between the actually received irradiance on a surface and the theoretically obtainable irradiance on the outer side of the atmosphere. The instantaneous clearness index is defined by instant global horizontal irradiance (I_{GHI}) and extraterrestrial irradiance (I_{ETI}) in the following:

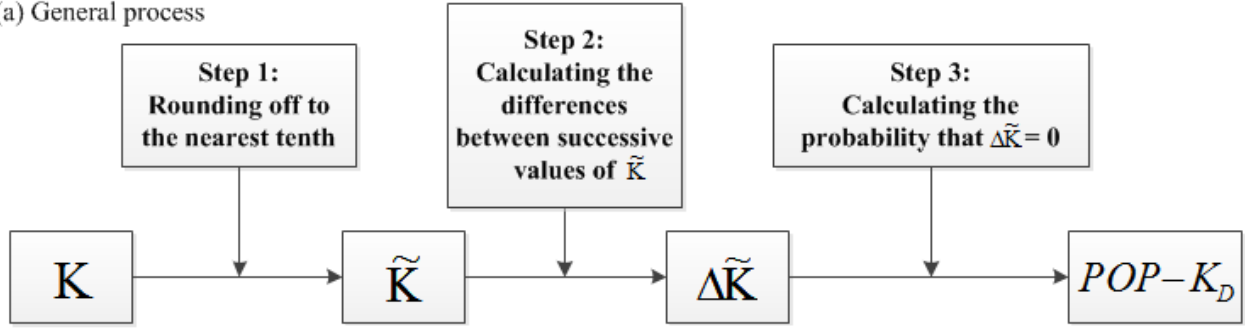
$$k = \frac{I_{GHI}}{I_{ETI}} = \frac{I_{GHI}}{I_0 E_0 \cos \theta_Z}, \quad (3.1)$$

where I_0 is the solar constant, E_0 is the eccentricity correction factor, and θ_Z is the zenith angle on a surface (Iqbal, 1983). A single index can directly compare to other indices regardless of temporal and spatial differences due to a consideration of E_0 and θ_Z in Eq. (3.1). A short-time behavior of I_{GHI} is fluctuated by rapid meteorological or geographical changes. Therefore, a variation of the instantaneous indices is the best indicator reflecting the fluctuation of solar irradiance. Similarly, K_D is achieved by a ratio between a daily integration of I_{GHI} and I_{ETI} like:

$$K_D = \frac{\int_{Day} (I_{GHI}) dt}{\int_{Day} (I_{ETI}) dt}. \quad (3.2)$$

K_D indicates a quantitative amount of daily earned solar irradiance. Since the integration cannot reflect the short-term fluctuation of solar irradiance, the stand-alone use of K_D has a limitation for the quality analysis.

(a) General process



K : a set of instantaneous clearness indices (during a day)

(b) Examples

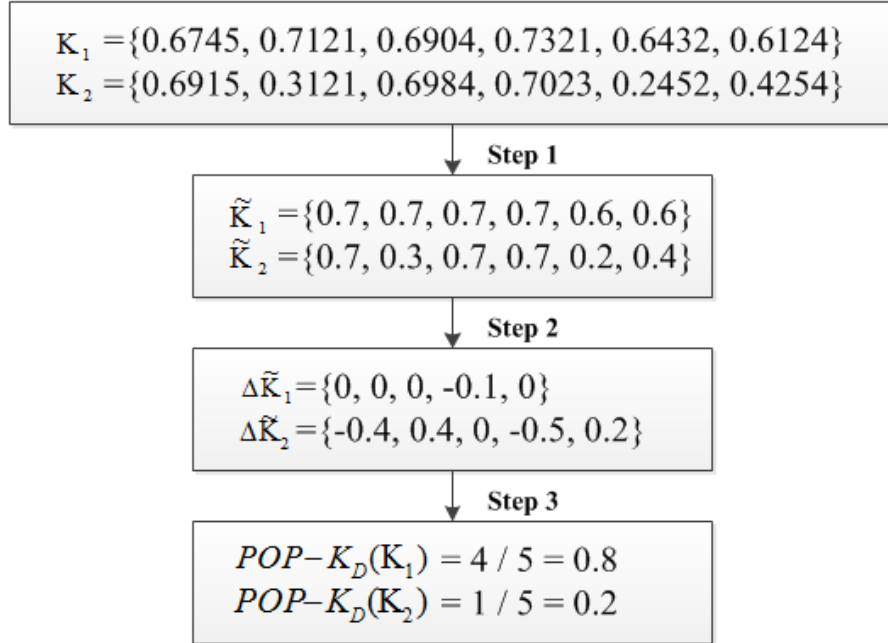


Fig. 3.1: Process and example for calculating POP- K_D .

Next, this study proposes a new measure for quality of daily solar irradiance: probability of persistence (POP- K_D). POP- K_D observes differences between neighboring instantaneous clearness indices and calculates a probability that the differences are equal to zero. Thus, the higher value of POP- K_D indicates the lower chance for fluctuation and steady irradiance capable of providing smoother electric power by its definition. Fig. 3.1 shows three steps to calculate POP- K_D (a) and examples with arbitrary two sets of instantaneous clearness indices (b). From the comparison of K_1 and K_2 in Fig. 3.1(b), the lower POP- K_D value of K_2 indicates the high fluctuating result.

For the characterization of daily sky conditions, the K-POP method observes the daily solar data through the grid by the 2-tuple as shown in Fig. 3.2. The x-axis and y-axis of the plot are K_D and POP- K_D respectively. This method has a strong advantage due to the quantitative and qualitative observation of daily solar irradiance. This advantage enables to distinguish different fluctuation levels under the similar amount of K_D . In addition, this method provides information about local climatic characteristics from the distribution's boundary shape and densities for a certain period, which indicate a snapshot and bias of local solar characteristics respectively.

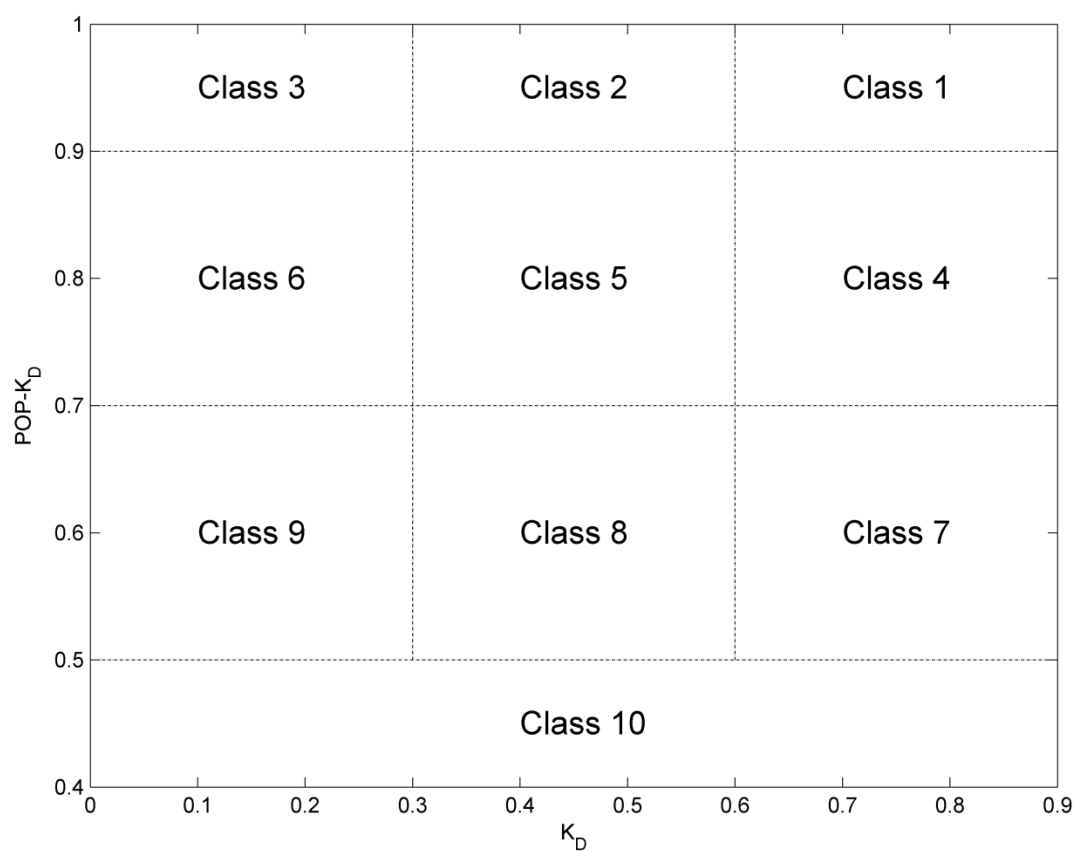


Fig. 3.2: The 10 classes defined by the proposed characterization and classification methods.

Table 3.2: Definition and description of the proposed 10 classes.

Class	K_D	POP- K_D	Description	Example
1	$K_D \geq 0.6$	$POP-K_D \geq 0.9$	High quantity & high quality; sunny and steady sky conditions for almost the entire day	Fig. 3a
2	$0.3 \leq K_D < 0.6$	$POP-K_D \geq 0.9$	Medium quantity & high quality; partly cloudy but sky conditions are relatively steady for most of the day	Fig. 3b
3	$K_D < 0.3$	$POP-K_D \geq 0.9$	Low quantity & high quality; overcast but sky conditions are relatively steady for most of the day	Fig. 3c
4	$K_D \geq 0.6$	$0.7 \leq POP-K_D < 0.9$	High quantity & medium quality; sunny but the sky conditions vary for part of the day	Fig. 3d
5	$0.3 \leq K_D < 0.6$	$0.7 \leq POP-K_D < 0.9$	Medium quantity & medium quality; partly cloudy and the sky conditions vary for part of the day	Fig. 3e
6	$K_D < 0.3$	$0.7 \leq POP-K_D < 0.9$	Low quantity & medium quality; cloudy and the sky conditions vary	Fig. 3f
7	$K_D \geq 0.6$	$0.5 \leq POP-K_D < 0.7$	High quantity & low quality; partly sunny with sky conditions varying significantly for most of the day	Fig. 3g

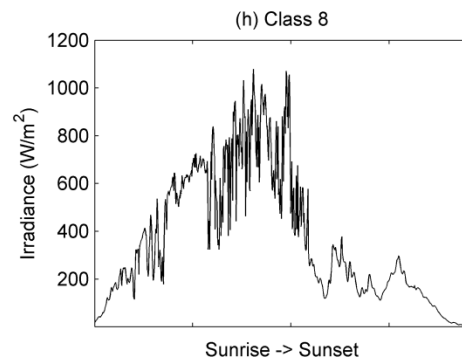
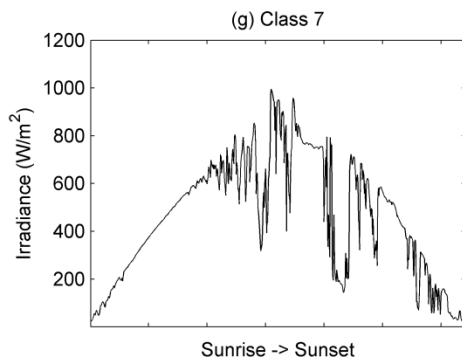
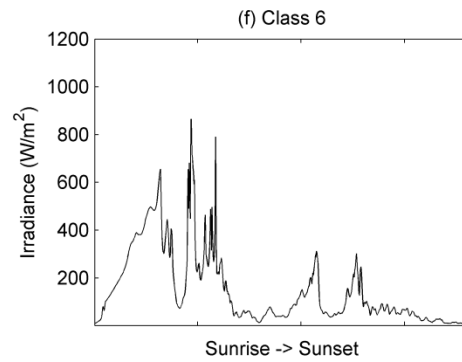
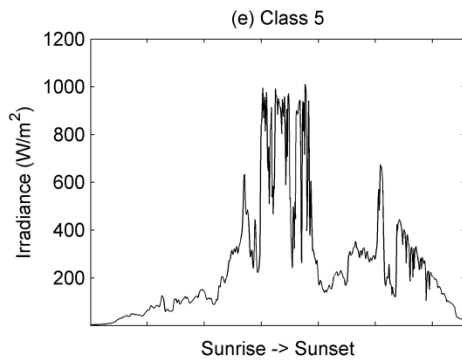
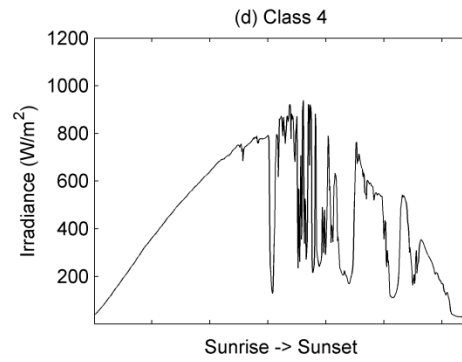
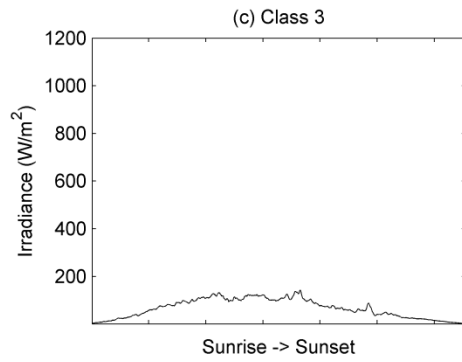
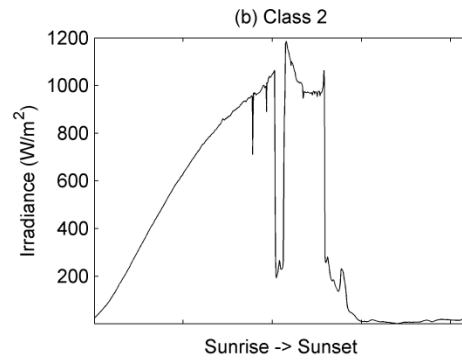
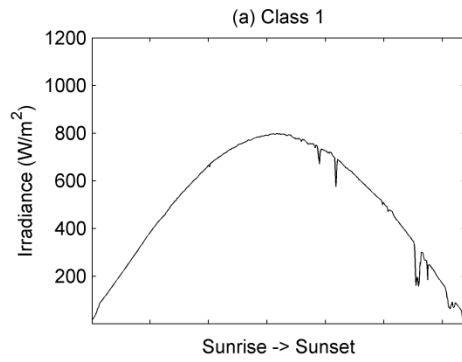
8	$0.3 \leq K_D < 0.6$	$0.5 \leq \text{POP} - K_D < 0.7$	Medium quantity & low quality; various degrees of cloudiness and the sky conditions varying significantly for most of the day	Fig. 3h
9	$K_D < 0.3$	$0.5 \leq \text{POP} - K_D < 0.7$	Low quantity & low quality; cloudy sky with high degree of fluctuations for the entire day	Fig. 3i
10	-	$\text{POP} - K_D < 0.5$	Very low quality; various degrees of cloudiness but high levels of fluctuations for the entire day	Fig. 3j

Table 3.3: Occurrence percentages and average values of K_D ($K_{D,ave}$) and POP- K_D (POP- $K_{D,ave}$) of the proposed 10 classes at three locations.

Class	SRRL BMS			UNLV			ORNL		
	%	$K_{D,ave}$	POP- $K_{D,ave}$	%	$K_{D,ave}$	POP- $K_{D,ave}$	%	$K_{D,ave}$	POP- $K_{D,ave}$
1	18.3	0.753	0.947	45.2	0.749	0.946	12.9	0.703	0.944
2	0.9	0.419	0.920	0.2	0.347	0.918	0.1	0.600	0.906
3	2.1	0.149	0.928	0.6	0.146	0.931	5.8	0.123	0.929
4	29.4	0.697	0.802	29.6	0.706	0.812	20.8	0.671	0.812
5	21.9	0.475	0.777	9.3	0.472	0.775	17.6	0.457	0.769
6	5.6	0.201	0.837	1.6	0.220	0.835	15.1	0.186	0.820
7	8.7	0.664	0.642	7.3	0.667	0.641	5.8	0.639	0.644
8	12.6	0.502	0.644	6.0	0.517	0.642	19.9	0.481	0.632
9	0.1	0.259	0.684	0.1	0.274	0.685	1.4	0.260	0.663
10	0.3	0.612	0.470	0.1	0.615	0.485	0.6	0.489	0.467

3.3.2. New classification method

This study also introduces a new classification method based on the grid by the K-POP method. The new classification method symmetrically cut part of the grid and divides the partial grid into 10 classes in rectangular as shown in Fig. 3.2. The x-axis was symmetrically divided into three typical classes (clear, partly overcast, and entirely overcast sky) using 0.3 and 0.6, similar to the previous quantity-based approaches (Maafi and Harrouni, 2003; Rahim et al., 2004; Harrouni et al., 2005; Badescu, 2008; Baharuddin et al., 2010). The boundaries of y-axis were 0.5, 0.7, and 0.9 based on the annual characterization results. As a result, each class has a specified quantity and quality information of daily solar irradiance. Table 3.2 provides boundaries of the 2-tuple and description about both quality and quantity of I_{GHI} for each class. Related to the proposed classes, Fig. 3.3 shows an example of daily changes of I_{GHI} from sunrise to sunset for all ten classes. In addition, Table 3.3 shows occurrence percentages and average values of the 2-tuple of the 10 classes at three locations. These average values represent the specified information of a given class at each location.



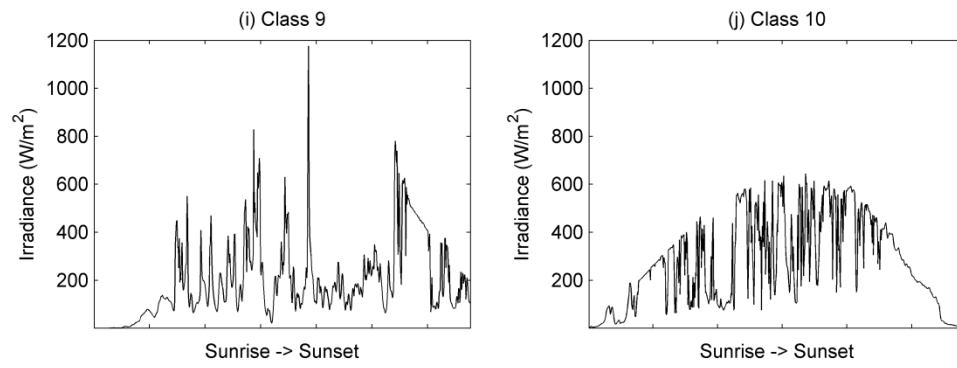


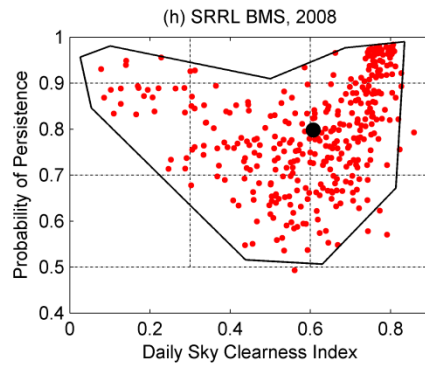
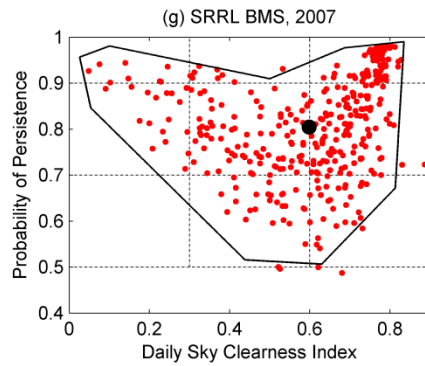
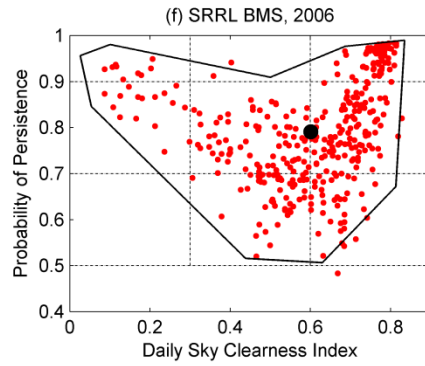
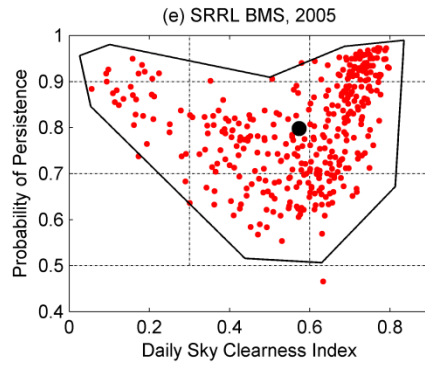
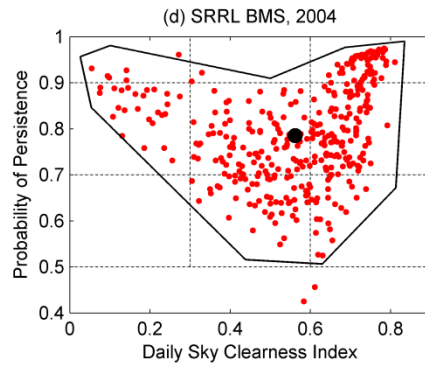
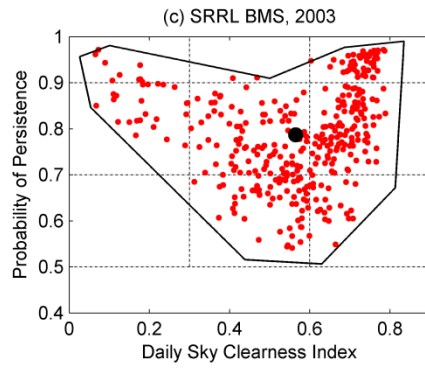
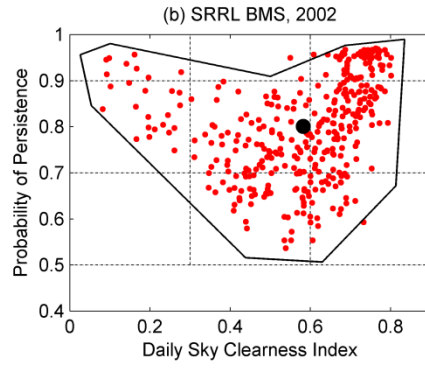
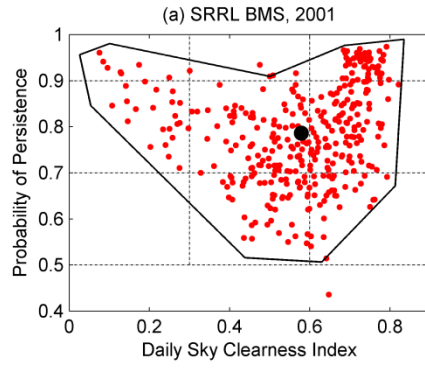
Fig. 3.3: Examples of the daily solar irradiance plot of each class from sunrise to sunset.

3.4. Results

3.4.1. Consistent patterns

Fig. 3.4 shows 11 annual patterns for the K-POP 2-tuple for the data in SRRL BMS from 2001 to 2011 (Fig. 3.4a – 4k) and the overall multi-year pattern that uses all the data in one plot (Fig. 3.4l). Fig. 3.4 demonstrates some significant results. First, almost all data in every plot of Fig. 3.4 can be enclosed by an irregular nonagon boundary shown in each plot. Although this boundary encloses an area that spans over most classes, a range of POP- K_D can be identified for each value of K_D . As shown in Table 3.4, the range of POP- K_D is the largest (from 0.51 to 0.96) when K_D is 0.63, and it is getting smaller as K_D increases and as K_D decreases. When the range of POP- K_D gets smaller, both the lower and upper limits of the ranges increase to higher POP- K_D values. When K_D reaches 0.84, the POP- K_D range is collapsed into a single value of 0.99. Similarly, when K_D reaches 0.03, the POP- K_D range is collapsed into a single value of 0.96. The irregular nonagon boundary indicates that the qualities of irradiance are generally high and more certain (smaller ranges of variation) at very high and at very low quantity levels, but the qualities are generally lower and vary within wider ranges for the rest of the quantity levels.

The distribution patterns of the data points are similar in all the plots in Fig. 3.4. To further illustrate this point, the annual average of the 2-tuple is computed and represented by the black dot in each plot. As shown in Fig. 3.4, the black dot representing the statistical “center” is consistently located at approximately the same location, around 0.8 for POP- K_D and around 0.6 for K_D .



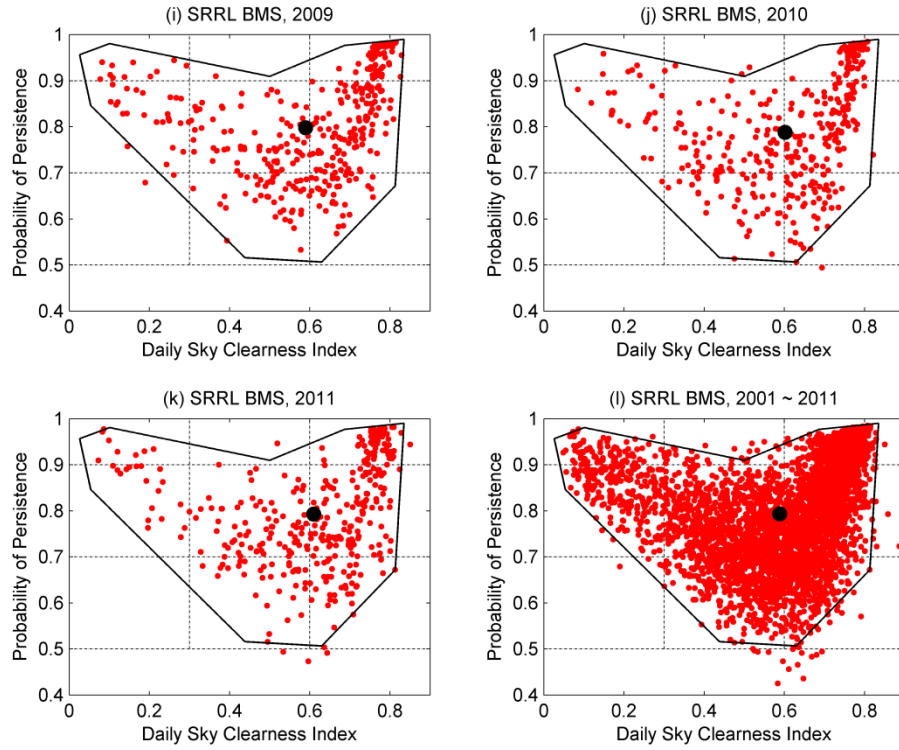


Fig. 3.4: Annual and multi-year plots for SRRL BMS (2001 ~ 2011).

Table 3.4: Range of POP- K_D for different values of K_D on the irregular nonagon boundary.

K_D	Range of POP- K_D
0.03	0.96
0.05	0.85 – 0.97
0.10	0.81 – 0.98
0.44	0.52 – 0.92
0.50	0.51 – 0.91
0.63	0.51 – 0.96
0.69	0.56 – 0.98
0.81	0.67 – 0.99
0.84	0.99

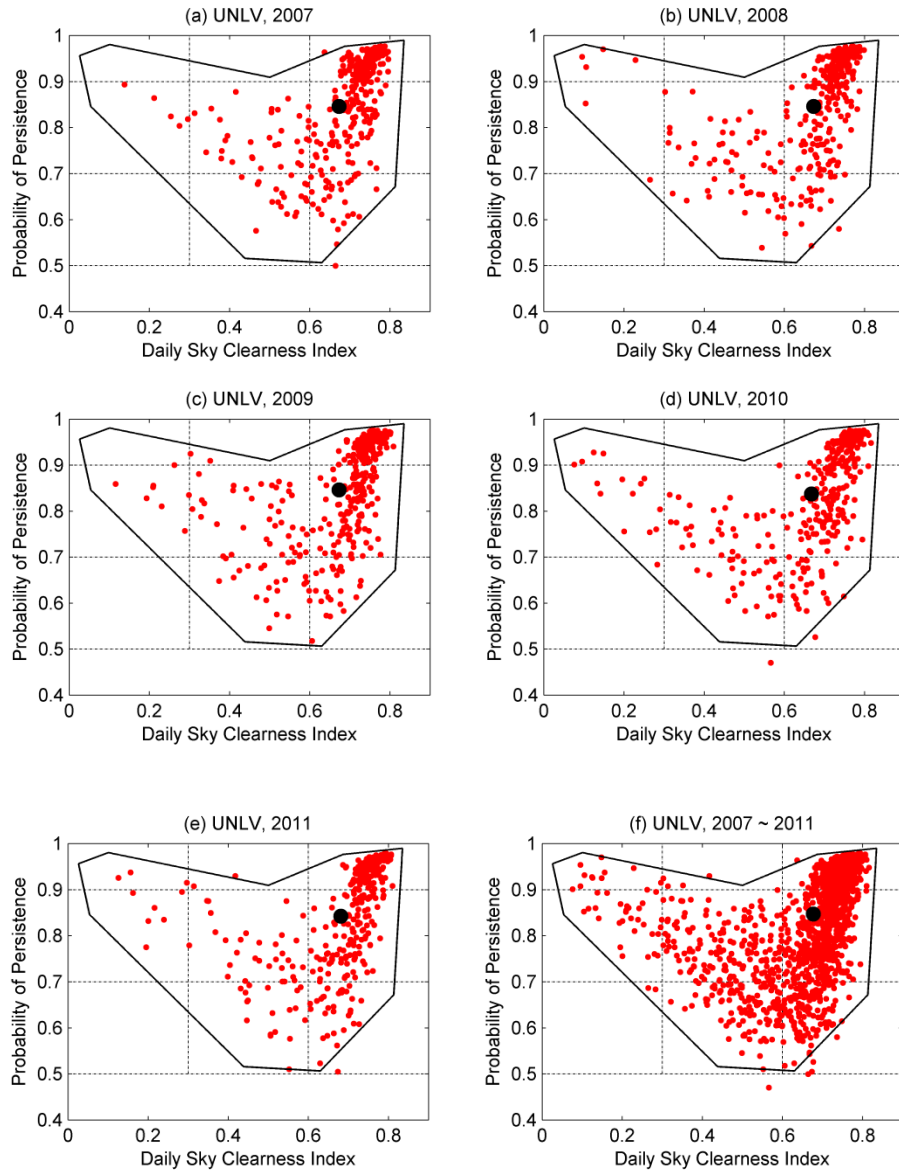


Fig. 3.5: Annual and multi-year plots for UNLV (2007 ~ 2011).

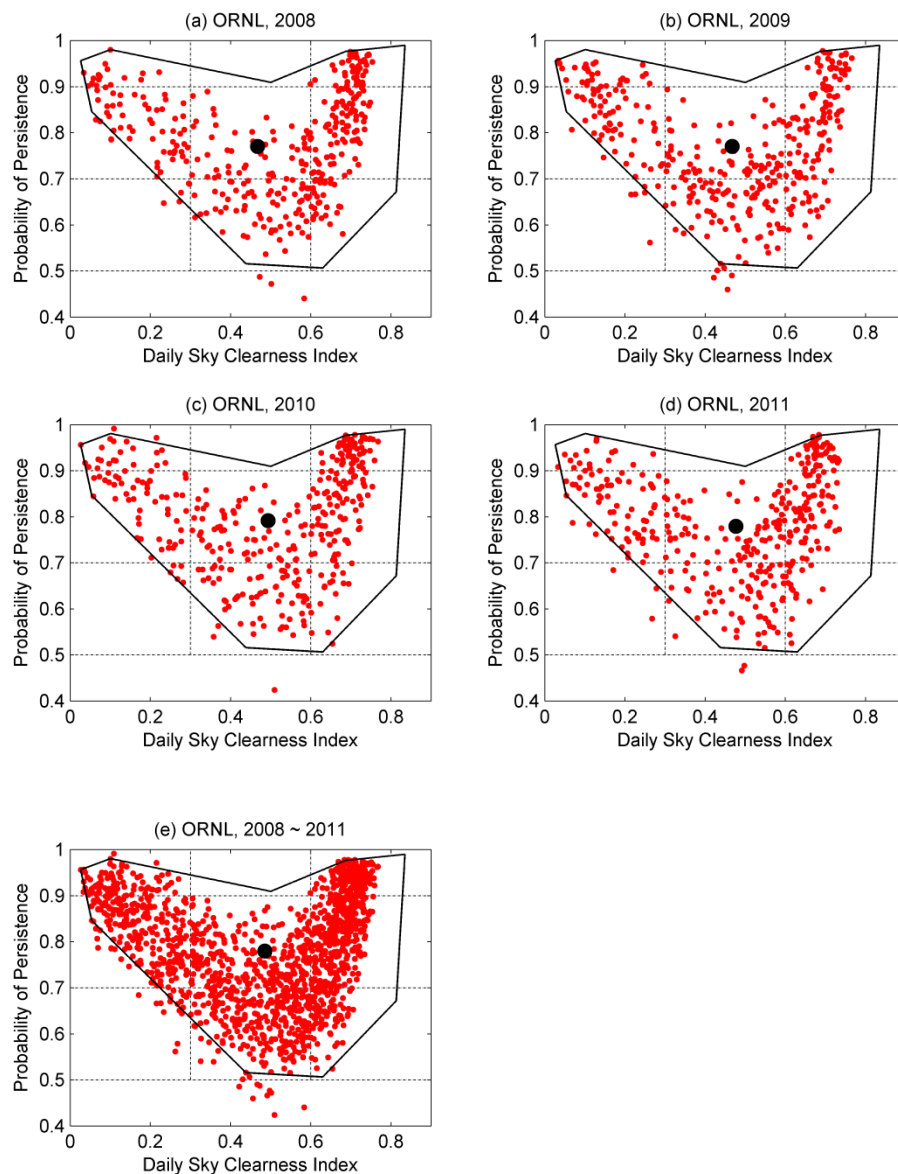


Fig. 3.6: Annual and multi-year plots for ORNL (2008 ~ 2011).

Fig. 3.5 and 3.6 show similar plots for the data in UNLV and ORNL, respectively. Most of the points in UNLV and ORNL are located inside of the irregular nonagon boundary like SRRL BMS. The centers of the 2-tuple in both UNLV and ORNL are annually located at the almost same positions.

Although the same boundary is applicable to all three locations, the three locations show different distribution patterns. These differences are expected by their different local climates and reflected by different population densities and center positions. More points in UNLV are distributed in the upper right corner of the plots compared to the plots in SRRL BMS and ORNL, and consequently the center of UNLV is weighted toward the upper right. It might be related to its desert climate. In ORNL, points are more or less evenly distributed in a “V”-shape fashion as shown clearly in Fig. 3.6 within the heart-shape boundary. The center of ORNL is moved toward the middle of the grid.

Fig. 3.7 shows the ranges of K_D and $POP-K_D$ of the 2-tuple at SRRL BMS, UNLV, and ORNL respectively. In each year, a dot in the middle represents the annual mean, and the range between the dot and edges represents the standard deviation. For both K_D (Fig. 3.7a) and $POP-K_D$ (Fig. 3.7b), all three locations exhibit statistical consistencies for the means and standard deviations. These annual consistencies suggest that one-year data may be sufficiently representative of the location’s annual climatic characteristics as far as the 2-tuple is concerned. In addition, Fig. 3.7 also verifies the aforementioned difference in distribution patterns at different locations. UNLV shows the larger mean values of the 2-tuple compared to the other two locations, and this corresponds to its weighted center toward the upper right as indicated in Fig. 3.5. The annual means of $POP-K_D$ in SRRL BMS and ORNL are similar (Fig. 3.7b), but the

means of K_D in ORNL are relatively lower (Fig. 3.7a). This explains that the center in ORNL is located in the middle of the grid as shown in Fig. 3.6.

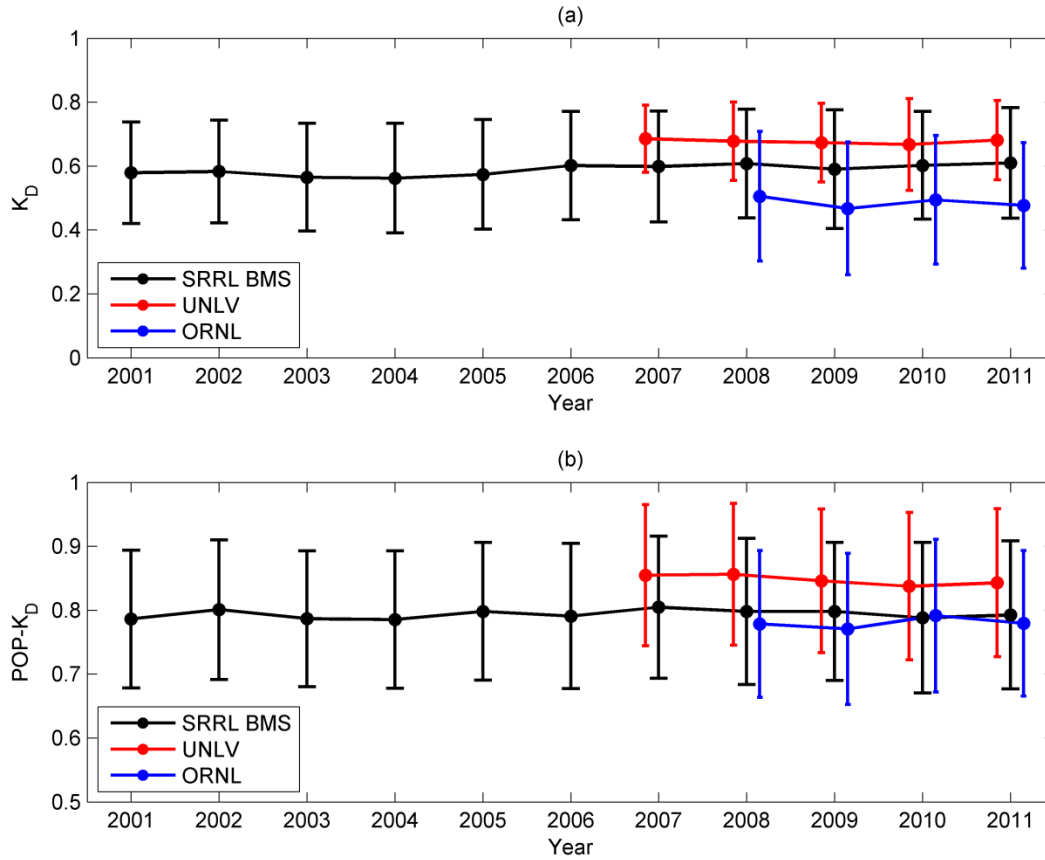


Fig. 3.7: Ranges (annual mean ± 1 standard deviation) for (a) K_D and (b) POP- K_D for 11 years at SRRL BMS, 5 years at UNLV, and 4 years at ORNL.

3.4.2. Identification of dominant sky conditions

Each class classified by the proposed classification method describes a distinct category of sky condition and may be considered a state of a Markov Model (Meyn et al., 2009). The transition of the sky condition from one category to another may be viewed as a transition from one state to another state. Table 3.5, 3.6, and 3.7 show the Markov transition matrices and frequencies of occurrence for the 10 classes for SRRL BMS, UNLV, and ORNL, respectively.

From the frequencies of occurrences for the 10 classes in SRRL BMS (Table 3.5), Class 1, 4, 5, and 8 have a range from 12.6% to 29.4%. The sum of these four classes takes more than 82%. In UNLV, Class 1 and 4 shows the significant dominances with about 45% and 30% respectively (Table 3.6). In addition, Class 5 approaches 10% here. These three classes together account for about 85% of the sky conditions. In ORNL (Table 3.7), Class 1, 4, 5, and 8 are dominant like SRRL BMS, but Class 6 also provide noticeable (15.1%) contribution. Class 1, 4, and 5 are the common dominant classes in all three locations.

Next, Table 3.5, 3.6, and 3.7 also show that the probabilities of transitions between the dominant classes are higher than those between other classes. For SRRL BMS (Table 3.5), when the given state is Class 1, the probabilities of transitioning to Class 1, 4, and 5 are 28.6%, 33.5% and 18.8%, respectively while the probabilities of transitioning other classes are less than 10%. Similar patterns are observed for Class 4 and 5 and in other locations. For UNLV (Table 3.6), the probability that Class 1 will remain at Class 1 even reaches 63.4%. These high transition probabilities between the dominant classes indicate that dominant states tend to transition among themselves.

Table 3.5: Transition matrix and frequency distribution of classes in SRRL BMS.

Class	Class										Occurrence Frequency
	1	2	3	4	5	6	7	8	9	10	
1	0.286	0.01	0.014	0.335	0.188	0.015	0.082	0.068	0	0.003	0.183
2	0.167	0.028	0.028	0.306	0.389	0	0.028	0.056	0	0	0.009
3	0.106	0.012	0.141	0.235	0.176	0.129	0.071	0.106	0.024	0	0.021
4	0.177	0.009	0.013	0.316	0.221	0.044	0.098	0.114	0	0.008	0.294
5	0.143	0.01	0.018	0.258	0.276	0.074	0.072	0.148	0	0.001	0.219
6	0.164	0.004	0.022	0.213	0.191	0.151	0.116	0.138	0	0	0.056
7	0.188	0.014	0.026	0.305	0.177	0.043	0.088	0.157	0.003	0	0.087
8	0.135	0.002	0.034	0.285	0.2	0.073	0.087	0.18	0.002	0.002	0.126
9	0.25	0	0	0	0	0	0.25	0.25	0	0.25	0.001
10	0.143	0	0	0.357	0.143	0	0.214	0.143	0	0	0.003

Table 3.6: Transition matrix and frequency distribution of classes in UNLV.

Class	Class										Occurrence Frequency
	1	2	3	4	5	6	7	8	9	10	
1	0.634	0	0.002	0.245	0.045	0.001	0.047	0.024	0	0.001	0.452
2	0	0	0	0.25	0.25	0	0.25	0.25	0	0	0.002
3	0	0	0.091	0.273	0.364	0	0	0.273	0	0	0.006
4	0.362	0.004	0.006	0.356	0.102	0.017	0.085	0.065	0.002	0.002	0.296
5	0.243	0.006	0.018	0.325	0.178	0.036	0.089	0.107	0	0	0.093
6	0.2	0	0.067	0.3	0.133	0.133	0	0.133	0.033	0	0.016
7	0.195	0	0	0.376	0.128	0.023	0.188	0.09	0	0	0.073
8	0.303	0.009	0	0.239	0.193	0.055	0.064	0.138	0	0	0.06
9	0	0	0	0	0	0.5	0.5	0	0	0	0.001
10	0	0	0	0.5	0	0	0	0.5	0	0	0.001

Table 3.7: Transition matrix and frequency distribution of classes in ORNL.

Class	Class										Occurrence Frequency
	1	2	3	4	5	6	7	8	9	10	
1	0.314	0	0.005	0.324	0.128	0.059	0.048	0.106	0.016	0	0.129
2	0	0	0	0	1	0	0	0	0	0	0.001
3	0.082	0	0.176	0.035	0.188	0.282	0.047	0.153	0.024	0.012	0.058
4	0.142	0.003	0.026	0.274	0.215	0.102	0.083	0.139	0.01	0.007	0.208
5	0.086	0	0.078	0.21	0.187	0.163	0.023	0.237	0.012	0.004	0.176
6	0.072	0	0.109	0.095	0.145	0.267	0.054	0.217	0.023	0.018	0.151
7	0.118	0	0.024	0.271	0.176	0.165	0.082	0.165	0	0	0.058
8	0.103	0	0.038	0.179	0.176	0.124	0.062	0.3	0.014	0.003	0.199
9	0	0	0.2	0.25	0.15	0.15	0.15	0.1	0	0	0.014
10	0.222	0	0	0.111	0.222	0.111	0.111	0.222	0	0	0.006

3.4.3. Monthly patterns of dominant classes

Fig. 8, 9, and 10 present the monthly patterns of the dominant classes for UNLV, SRRL BMS, and ORNL, respectively based on multiyear averages. As shown in Fig. 8, Class 1 is the most dominant class for most of the year, especially in the months from May to October. Class 4 is more dominant from late winter to early spring (February to April). Class 5 has a noticeable presence all year around.

As shown in Fig. 3.9, the monthly patterns in SRRL BMS are more complicated than those in UNLV. Class 4 is most dominant for most of the year. Class 5 shows stronger presence in the summer while Class 1 shows stronger presence from late summer to early spring. Class 8 has a noticeable presence all year around.

Fig. 3.10 shows that ORNL has the most complicated monthly patterns with the five dominant classes: Class 1, 4, 5, 6, and 8. Each class has its own seasonal pattern in such a way that the most dominant class is sky condition changes almost every month.

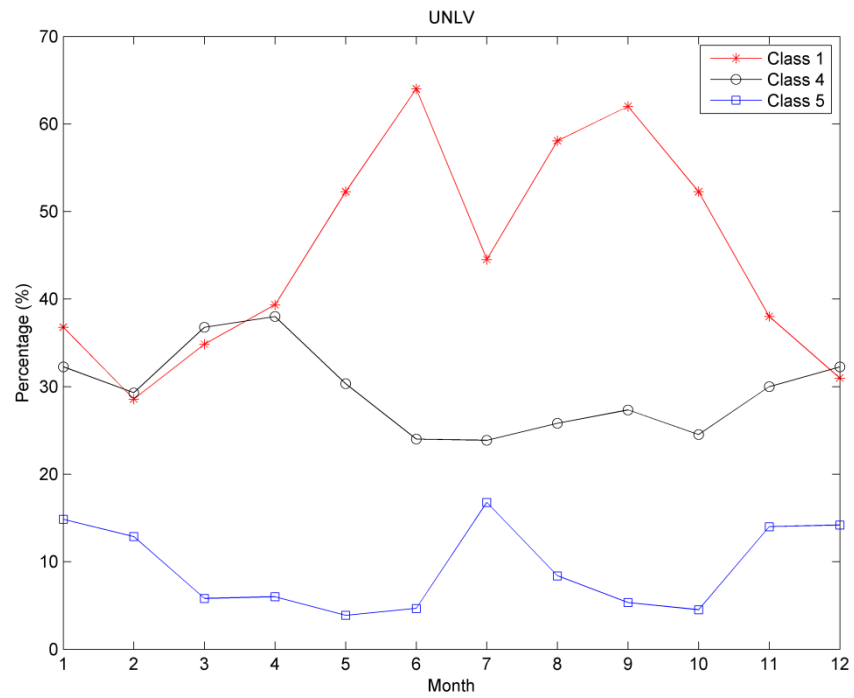


Fig. 3.8: Monthly patterns of Class 1, Class 4, and Class 5 in UNLV.

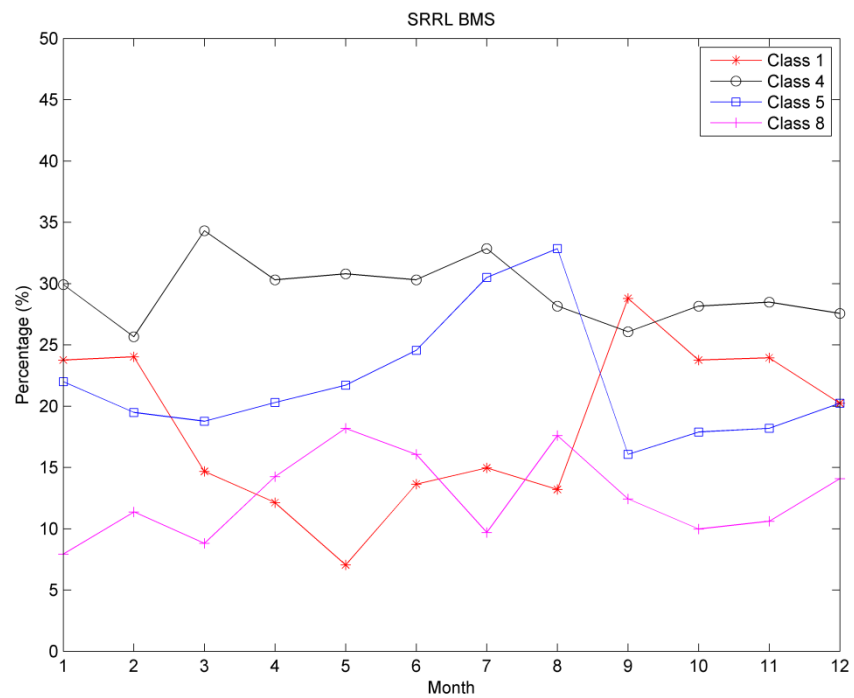


Fig. 3.9: Monthly patterns of Class 1, Class 4, Class 5, and Class 8 in SRRL BMS.

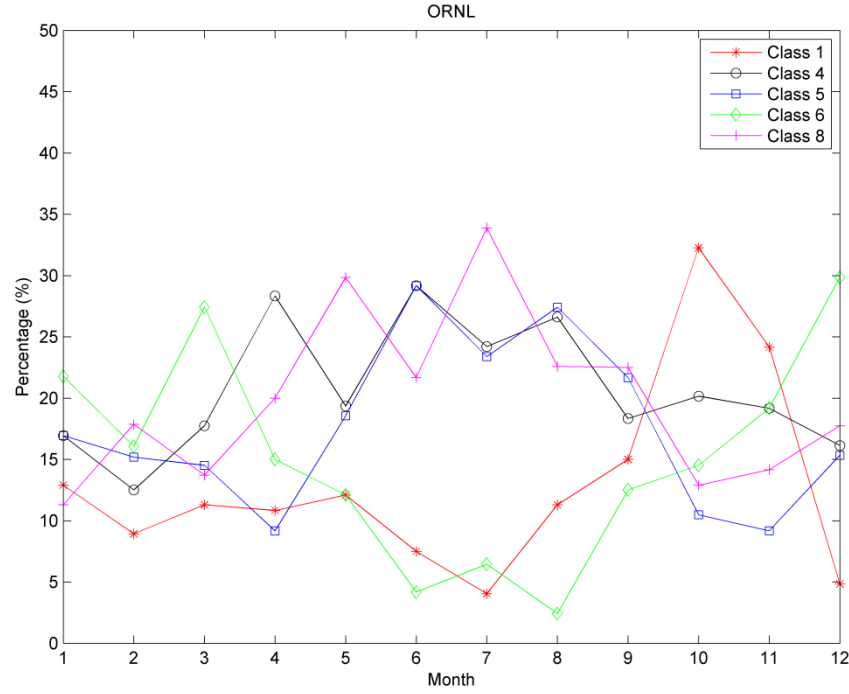


Fig. 3.10: Monthly patterns of Class 1, Class 4, Class 5, Class 6, and Class 8 in ORNL.

3.4.4. Seasonal patterns of K_D and POP- K_D

For seasonal investigations, this study uses the approach introduced in the study by Bendt et al. (1981) that divides 12 months into 3 groups: Summer (May, June, July, and August), Spring&Fall (March, April, September, and October), and Winter (January, February, November, and December). Fig. 3.11, 3.12, and 3.13 show the cumulative frequency distributions for K_D and POP- K_D for the 3 seasonal groups for SRRL BMS, UNLV, and ORNL, respectively. For each location, the first plot (Fig. 3.11a, 3.12a, and 3.13a) depicts the three-dimensional cumulative frequency distribution for the K_D and POP- K_D 2-tuple. The second plot (Fig. 3.11b, 3.12b, and 3.13b) is the two-dimensional projection of the three-dimensional distribution plot but with only K_D . These plots agreed well with the results established by Liu and Jordan (1960). The

third plot (Fig. 3.11c, 3.12c, and 3.13c) is the two-dimensional projection of the three-dimensional distribution plot with only POP- K_D . The relationship between K_D and POP- K_D is illustrated in the fourth plot (Fig. 3.11d, 3.12d, and 3.13d). The three-dimensional distribution demonstrates the fraction of time, which seasonally K_D and POP- K_D are equal to or less than certain values. For example, suppose K_D and POP- K_D on a certain day in June, Golden, are 0.64 and 0.80 respectively. Since June belongs to Summer, the percentage that K_D and POP- K_D are equal to or less than the give values of 0.64 and 0.80 respectively is approximately 59% according to the Summer curve in Fig. 3.11a. This value (59%) can be also verified by the two-dimensional distributions: the Summer curve of K_D at 0.64 (Fig. 3.11b) and that of POP- K_D at 0.80 (Fig. 3.11c).

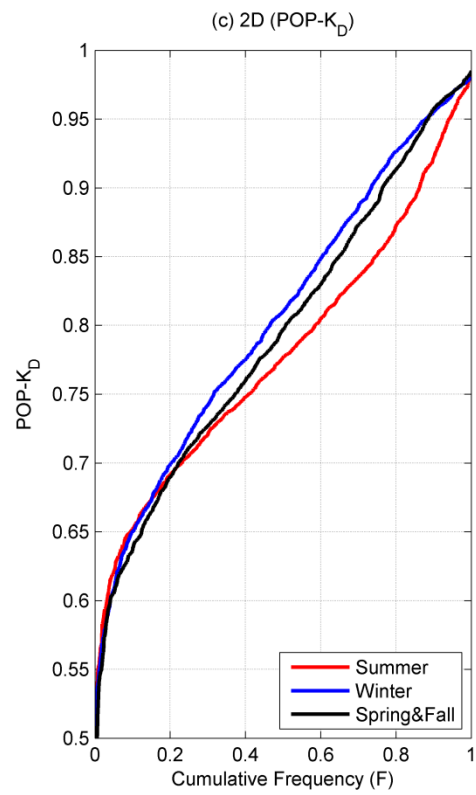
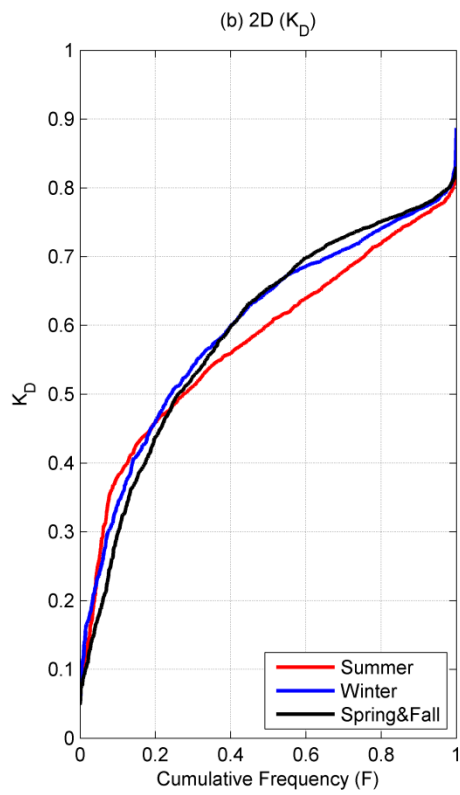
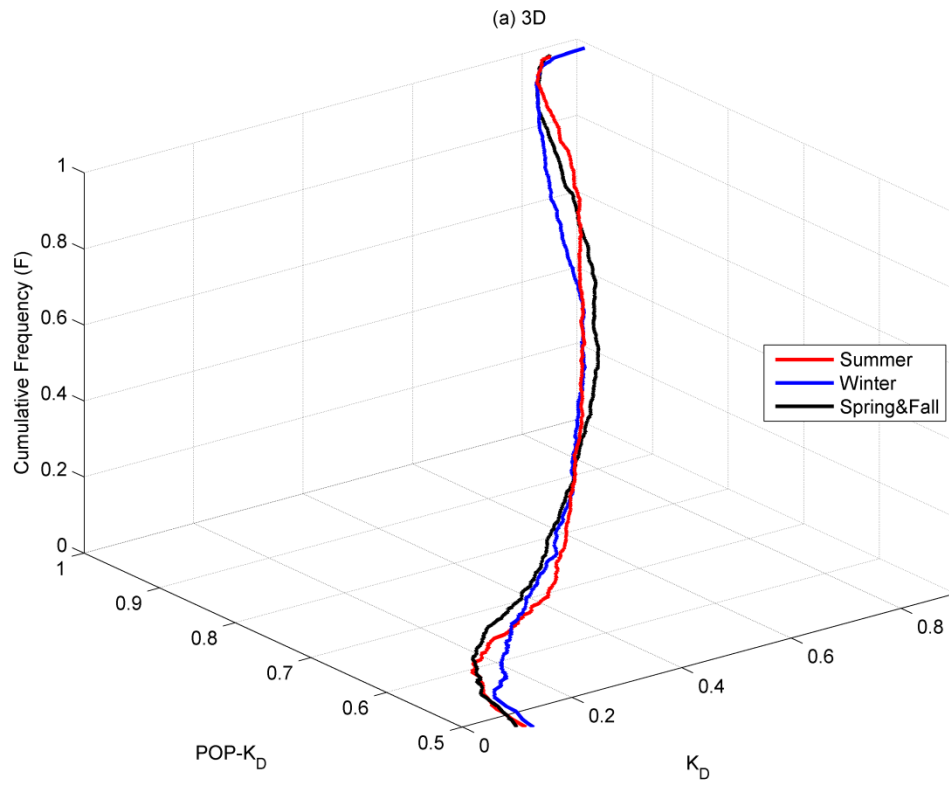
Table 8 summarizes the seasonal trends of K_D (part b), the seasonal trends of POP- K_D (part c), and the seasonal relationship between the two tuple (part d) for each location. $K_{D\text{-season}}$ and $POP\text{-}K_{D\text{-season}}$ indicate the corresponding seasonal trend of K_D and POP- K_D , respectively. SRRL BMS shows the least distinguishable seasonal relationship between K_D and POP- K_D since the three seasonal curves of Fig. 3.11d are closely tangled. This explains that similar patterns of the quality are expected regardless of the seasons here. In contrast, ORNL reveals relatively the most distinguishable seasonal relationship between the 2-tuple since the three seasonal curves of Fig. 3.13d are far away. Thus, the overall quality in summer is expected to be worse than that in winter at this location.

Although different locations show different seasonal patterns, the relationship between the seasonal trends of quantity (part b) and quality (part c) reveals similar patterns regardless of the location. As compared and described in Table 8, the seasonal trends of the 2-tuple display similar patterns for the same location; the seasonal curves of Summer for SRRL BMS, Winter

for UNLV, and Summer for ORNL are different or located downwards compared to the other two seasons for both 2-tuple.

Table 3.8: Comparison of seasonal trends for K_D (part b), $POP-K_D$ (part c), and K_D & $POP-K_D$ (part d) in Fig. 3.11, 3.12, and 3.13.

Location	Figure	Observed Trends
SRRL BMS	Fig. 3.11b	$K_{D-Spring\&Fall} \approx K_{D-Winter}$
	Fig. 3.11c	$POP-K_{D-Spring\&Fall} \approx POP-K_{D-Winter}$
	Fig. 3.11d	$POP-K_D$ for every K_D value is about the same for Winter, Summer, and Spring&Fall.
UNLV	Fig. 3.12b	$K_{D-Winter} < K_{D-Spring\&Fall} \approx K_{D-Summer}$
	Fig. 3.12c	$POP-K_{D-Winter} < POP-K_{D-Spring\&Fall} < POP-K_{D-Summer}$
	Fig. 3.12d	For almost every K_D value, $POP-K_D$ is higher in Winter than in Summer or Spring&Fall.
ORNL	Fig. 3.13b	$K_{D-Spring\&Fall}$ and $K_{D-Winter}$ are similar but $K_{D-Summer}$ is different.
	Fig. 3.13c	$POP-K_{D-Summer} < POP-K_{D-Spring\&Fall} \approx POP-K_{D-Winter}$
	Fig. 3.13d	For every K_D value between 0.1 and 0.7, $POP-K_D$ is higher in Winter, lowest in Summer, and in between in Spring&Fall.



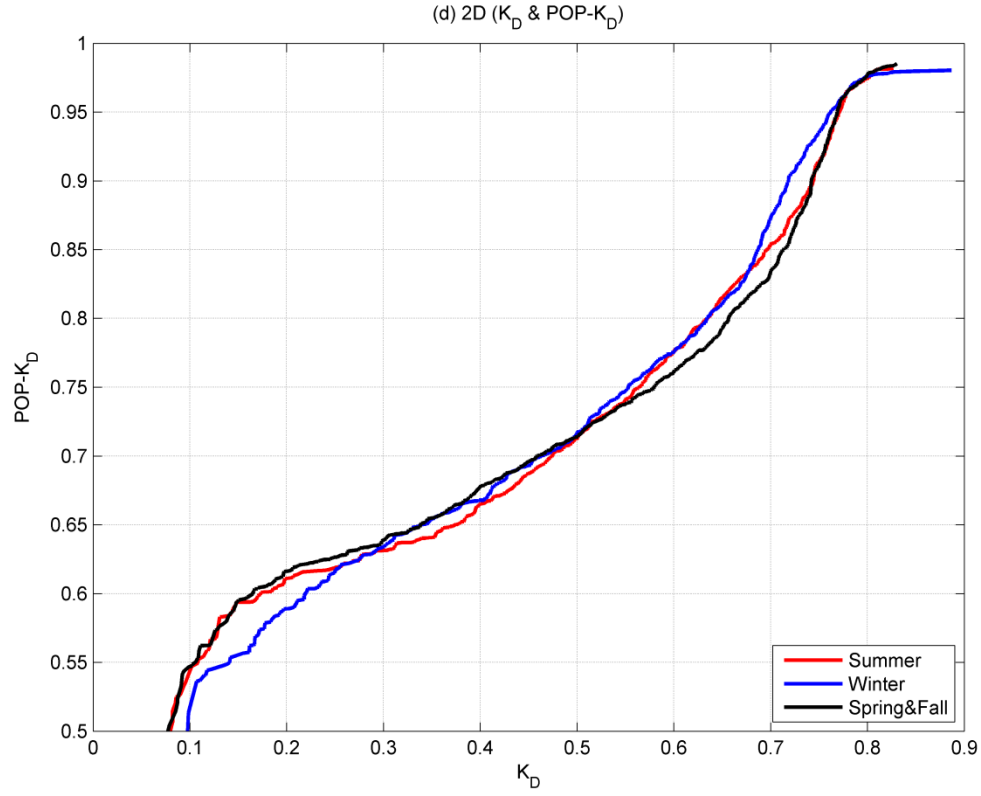
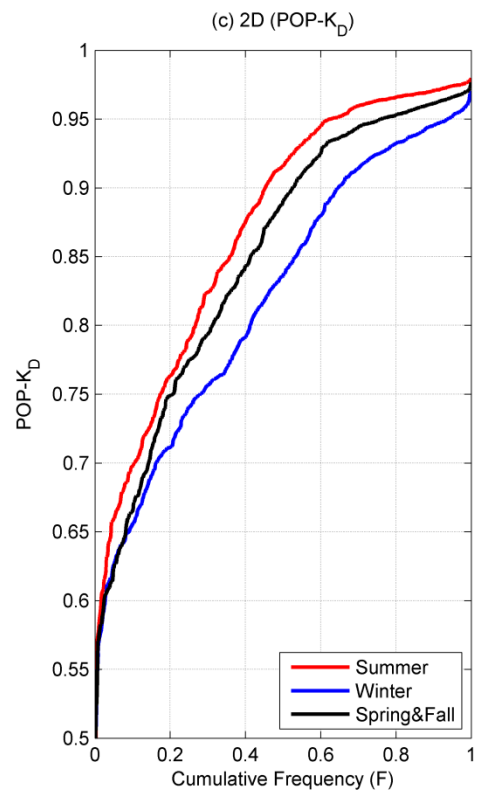
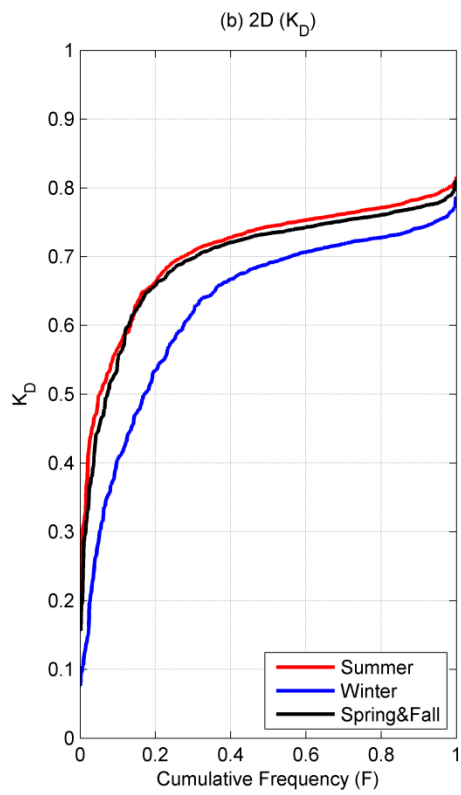
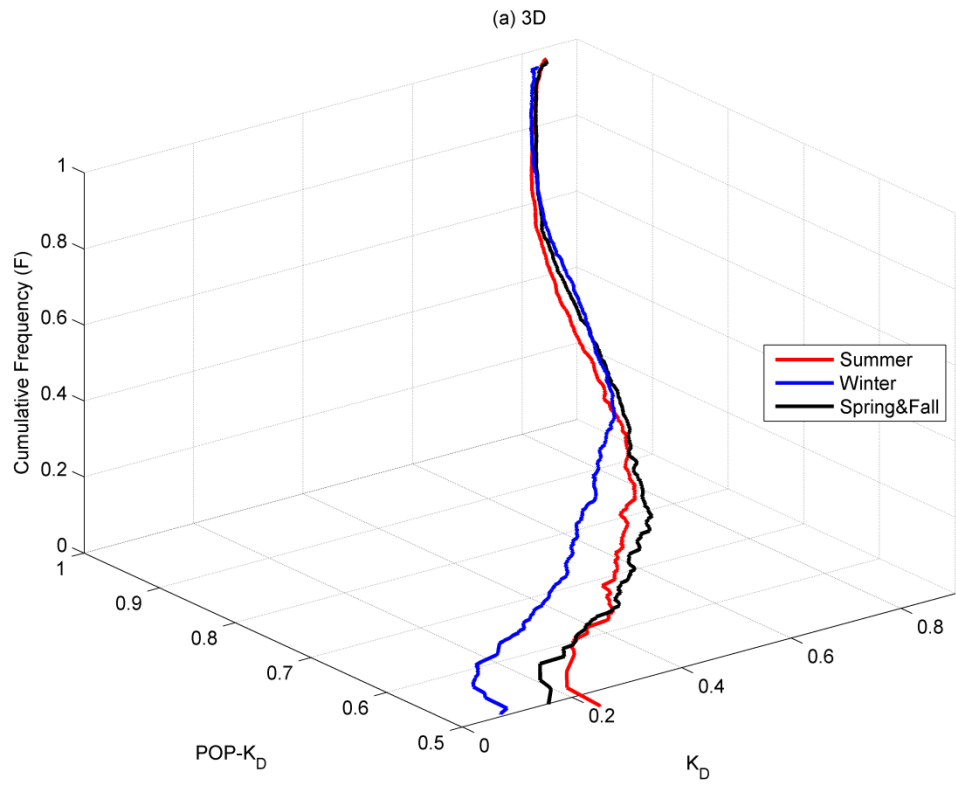


Fig. 3.11: Seasonal cumulative frequency distributions of (a) K_D and POP- K_D , (b) K_D , and (c) POP- K_D and (d) seasonal relationship between K_D and POP- K_D in SRRL BMS.



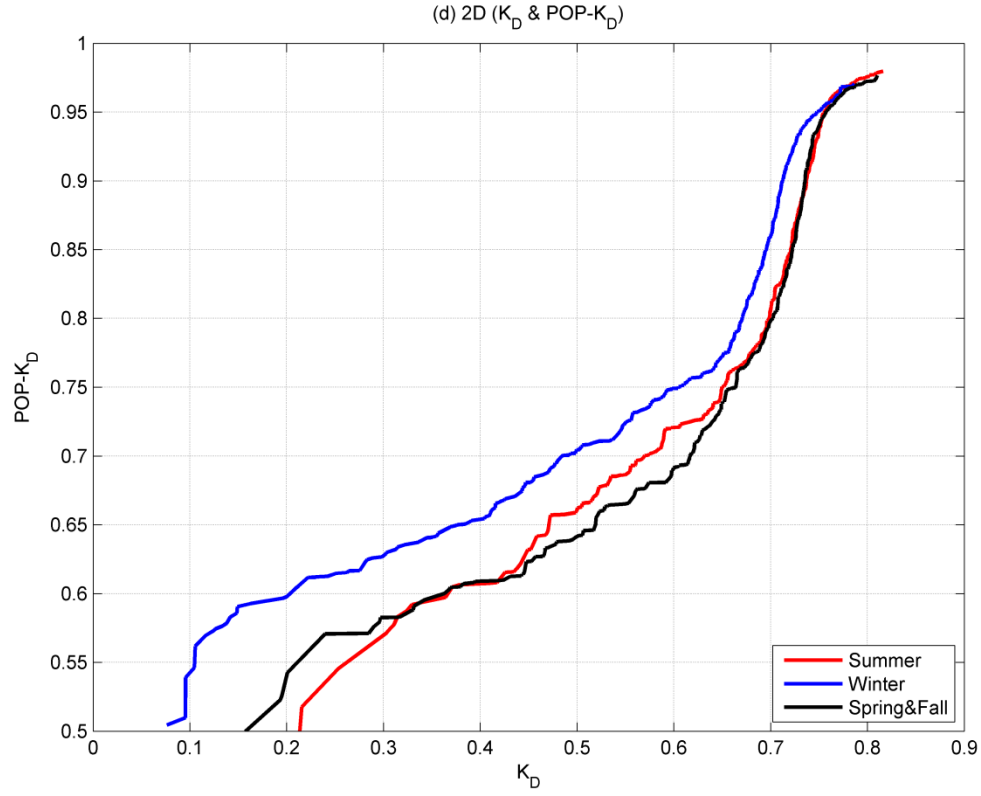
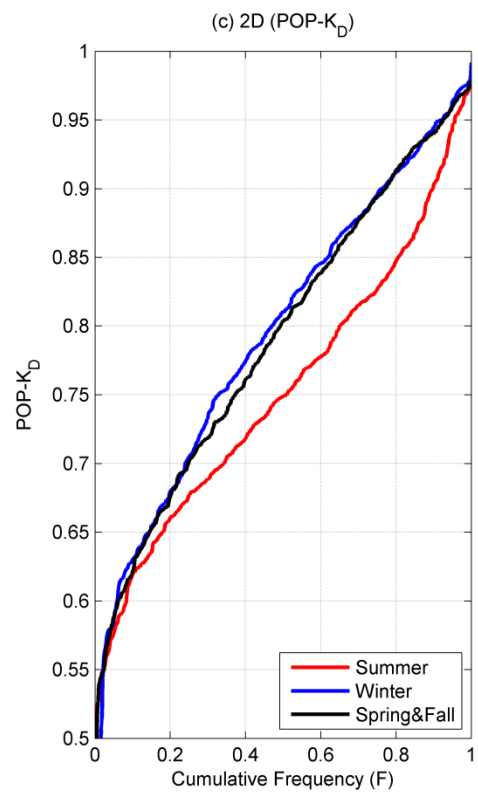
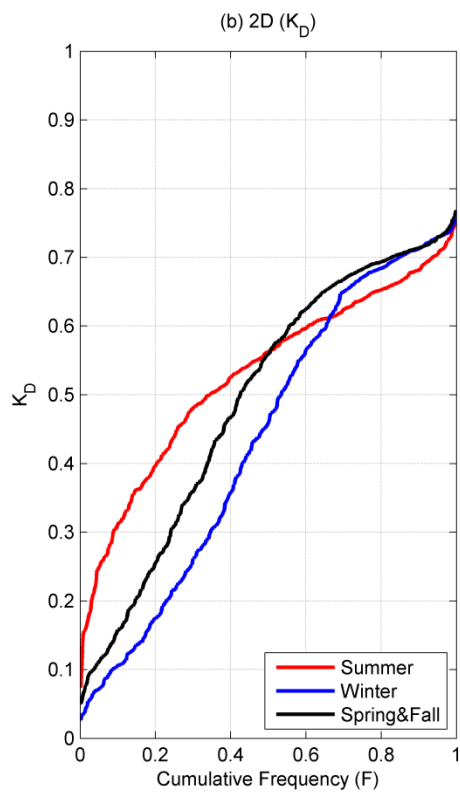
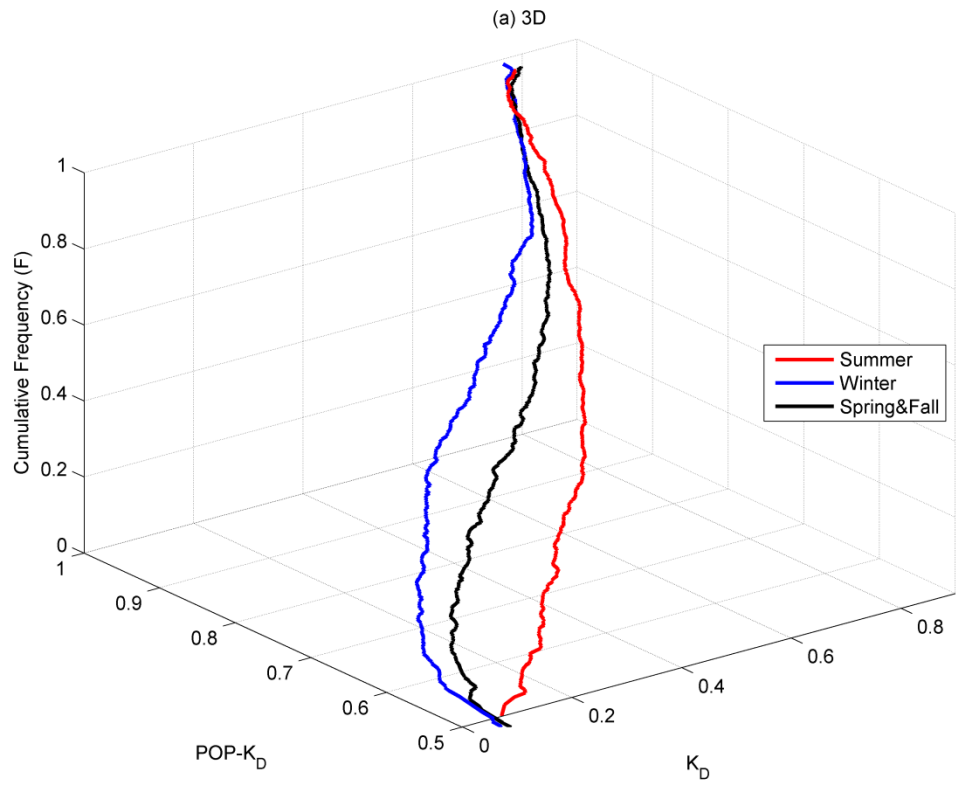


Fig. 3.12: Seasonal cumulative frequency distributions of (a) K_D and POP- K_D , (b) K_D , and (c) POP- K_D and (d) seasonal relationship between K_D and POP- K_D in UNLV.



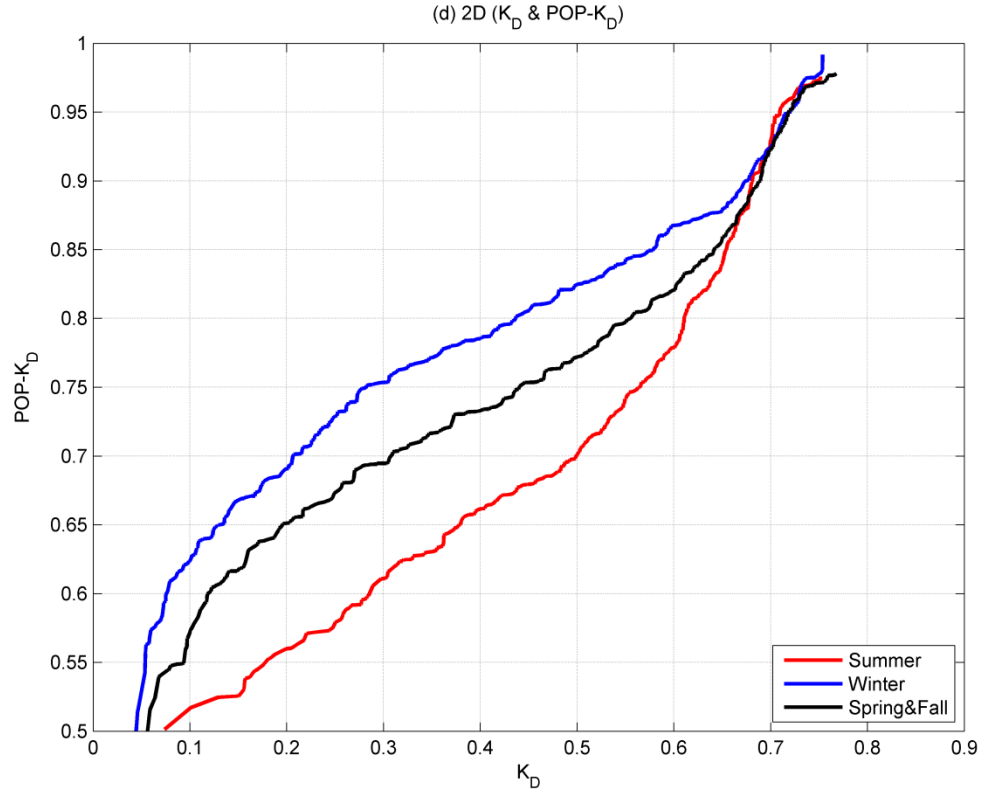


Fig. 3.13: Seasonal cumulative frequency distributions of (a) K_D and POP- K_D , (b) K_D , and (c) POP- K_D and (d) seasonal relationship between K_D and POP- K_D in ORNL.

3.5. Generation of K_D and POP- K_D sequences

This study provides a method to generate sequences for quantity (K_D) and quality (POP- K_D) of daily irradiance. This method originates from Aguiar et al. (1988), generating sequences of daily irradiance values based on a library of Markov Transition Matrices. K_D has a significant correlation with only preceding K_D values, and its probability density function for a certain period is only related with the average values of K_D for the period (Olseth and Skartveit, 1984; Aguiar et al., 1988). According to Aguiar et al. (1988), the synthetic sequence of K_D is useful for an accurate optimization of storage size for the solar-based generation system. It also helps to know the frequency of occurrence of available daily irradiance although there is a limitation for achieving the correct sequence. Similarly, the synthetic sequence of POP- K_D can be also useful for the optimization of the storage size and the occurrence frequencies of possible quality of daily irradiance. In addition, it may guide a better protection of electric devices, such as switching devices, connected to the solar-based generation system since POP- K_D is the measure for the fluctuation of the daily irradiance. For both K_D and POP- K_D , the synthetic sequences may substitute observed data for many years when there exists limited observed data containing a few years or only monthly averages.

Before generating the sequences, a library of transition matrices for K_D and POP- K_D are required. First, both K_D and POP- K_D are divided by 10 different classes depending on their monthly average values. Suppose the monthly average of K_D and POP- K_D are \overline{K}_D and $\overline{POP \cdot K}_D$ respectively. The 10 classes for K_D are adjusted from the Aguiar et al. (1988) work and defined; class 1 is when $\overline{K}_D \leq 0.40$, class 2 – 8 are increased in steps of 0.05 from 0.40 to 0.80, and class 10 is when $\overline{K}_D > 0.80$. Similarly, the 10 classes for POP- K_D are defined based

on $\overline{POP \cdot K_D}$ from 0.70 to 0.90 with 0.025 steps. Next, group K_D or $POP \cdot K_D$ values in each month into the 10 classes based on the corresponding $\overline{K_D}$ or $\overline{POP \cdot K_D}$ of this month, and find the maximum ($K_{D,max}$ and $POP \cdot K_{D,max}$) and minimum value ($K_{D,min}$ and $POP \cdot K_{D,min}$) in each class. Table 3.9 shows the maximum, minimum, and the number of months for each class using the data in SRRL BMS (2001 – 2011), UNLV (2007 – 2011), and ORNL (2008 – 2011). Then, each class is quantized by 10 states with a quantization level (L);

$$L = (K_{D,max} - K_{D,min})/10 \quad or \quad L = (POP \cdot K_{D,max} - POP \cdot K_{D,min})/10. \quad (3.3)$$

Consequently, each class generates a 10-by-10 transition matrix based on the grouped daily values, and the matrices of all classes for each tuple become the library.

Table 3.9: Maximum, minimum, and number of months of K_D and $POP \cdot K_D$ in 10 classes for generating a library of MTMs.

Class	1	2	3	4	5	6	7	8	9	10
$K_{D, min}$	0.0371	0.0264	0.0452	0.0431	0.0559	0.0728	0.1565	0.303	0.3641	0
$K_{D, max}$	0.7536	0.7658	0.7538	0.8277	0.8303	0.8868	0.8267	0.8161	0.8093	0
# of months	4	12	7	41	68	57	31	17	3	0
$POP \cdot K_{D, min}$	0.5390	0.5013	0.4595	0.4240	0.4558	0.3901	0.4401	0.5227	0.4992	0.5177
$POP \cdot K_{D, max}$	0.9386	0.9613	0.9851	0.9837	0.9841	0.9831	0.9916	0.9789	0.9798	0.9796
# of months	1	2	16	44	63	39	48	10	8	9

From the library, generating a daily value of K_D or POP- K_D begins with selecting one transition matrix (A) of K_D or POP- K_D depending on its allocated monthly average. Next, a random number R is generated between 0 and 1. Suppose that a previous daily value is state m . From the columns of the m th row in A , sum elements from the first column until the summed value is larger than or equal to R . Suppose that the stopped column index is n . Finally, the generated daily value is decided by the corresponding value of R calculated by a linear interpolation between the range of state n and the column elements of n and $n+1$ in A .

Fig. 3.14 and 3.15 show the observed and synthetic sequences of K_D and POP- K_D respectively for an entire year of 2011 at all three locations. Providing only monthly averages of the 2-tuple for each location, the 1-year synthetic sequences are generated based on the MTM libraries of the 2-tuple. All 6 pairs of the synthetic and observed sequences in Fig. 3.14 and 3.15 are tested using two parameters, Kolmogorov-Smirnov test Integral (KSI) and OVER proposed by Espinar et al. (2009), with 99% confidence level. Unlike the conventional Kolmogorov-Smirnov test, which observes a maximum difference between two cumulative frequency distributions of the sequence pairs (Corder and Foreman, 2009), the KSI calculates a percentage of the differences over the entire range while OVER provides an excessive percentage if the differences exceed the threshold boundary of the conventional K-S test (Espinar et al., 2009). According to Perez et al. (2010), the International Energy Agency Solar Heating and Cooling (IEA SHC) program Task 36 for performance benchmarking encourages the use of these parameters. Fig. 3.16 shows the distributional differences of the 6 pairs over all ranges, and the test results of KSI and OVER are presented in Table 3.10. The test results indicate that the KSI of all 6 pairs are relatively small (15.00% ~ 27.87%) and all OVER values are zeros except the pair of K_D in ORNL. However, the OVER of this pair was significantly small, 0.03%. These

results consequently explain similarities between the synthetic and observed sequences of the 2-tuple for all locations. Table 3.11 compares the annual averages of the 6 pairs. This also demonstrates the similarities of the two sequences due to the significantly small differences between the pairs (0.19% – 1.34%). Both the extended K-S test results and comparisons of the annual averages indicate a strength of the synthetic sequences generated by the libraries for the 2-tuple. Table 3.12 compares the occurrence percentages of the 10 classes based on the observed multi-year data and the average of 100 generated synthetic sequences at three locations. More work is planned to improve the proposed method.

Table 3.10: KSI and OVER results of synthetic and observed sequences of K_D and POP- K_D in Fig. 3.14 and 3.15.

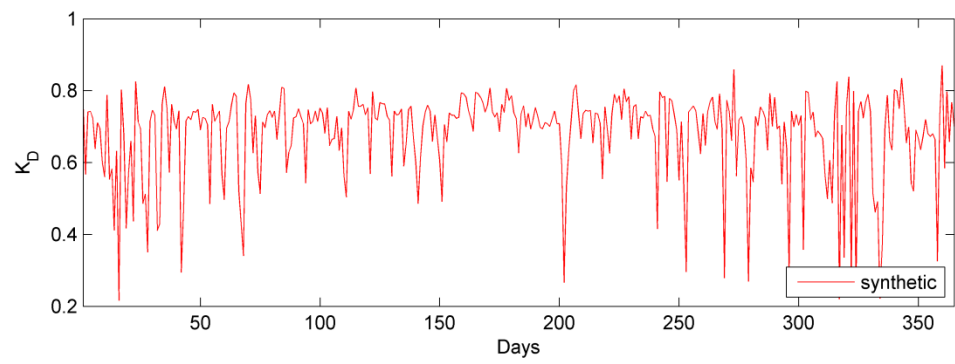
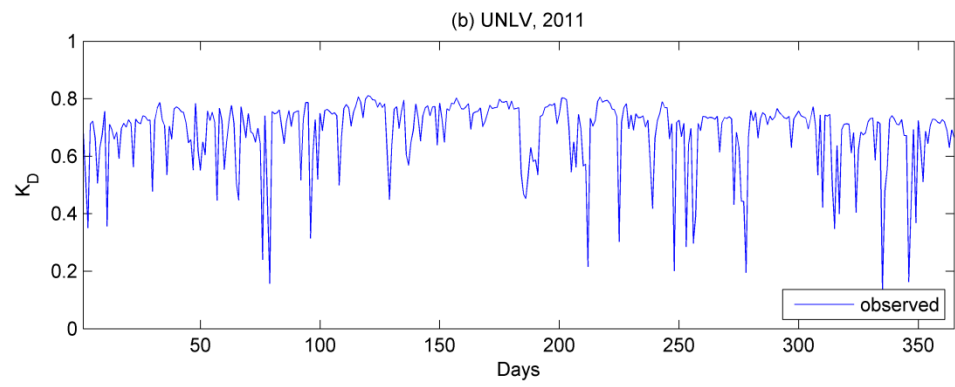
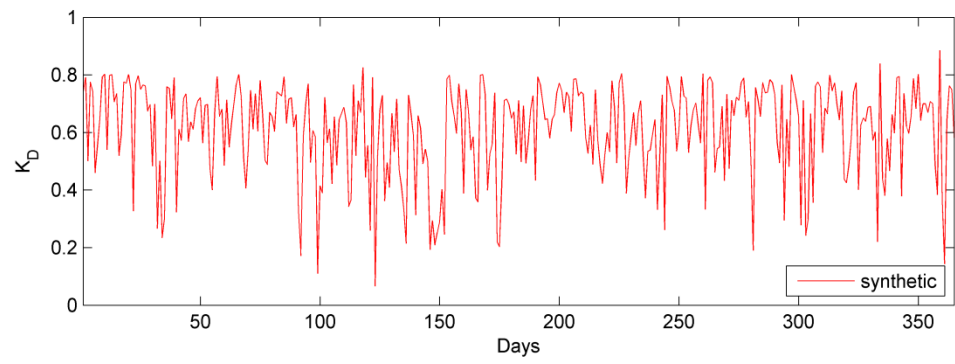
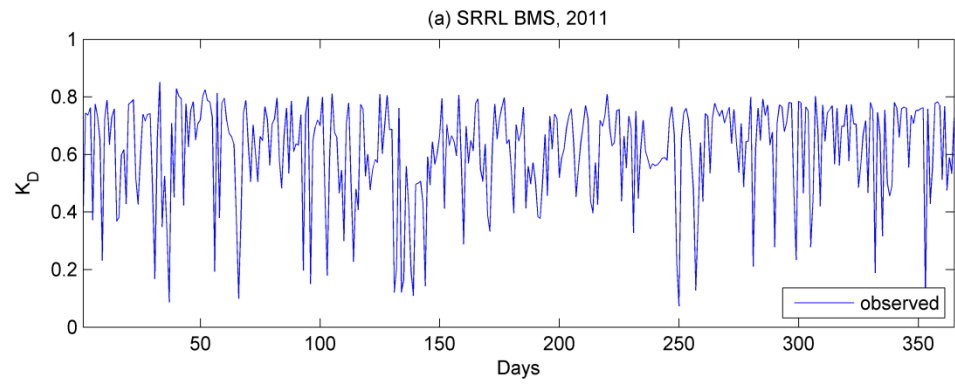
Location	K_D		POP- K_D	
	KSI (%)	OVER (%)	KSI (%)	OVER (%)
SRRL BMS	16.35	0.00	27.87	0.00
UNLV	15.00	0.00	23.15	0.00
ORNL	21.80	0.03	23.35	0.00

Table 3.11: Annual averages of synthetic and observed sequences of K_D and POP- K_D in Fig. 3.14 and 3.15.

Location	K_D			POP- K_D		
	Synthetic Average	Observed Average	Difference (%)	Synthetic Average	Observed Average	Difference (%)
SRRL BMS	0.6085	0.6097	0.19	0.7990	0.7924	0.84
UNLV	0.6790	0.6812	0.32	0.8493	0.8428	0.77
ORNL	0.4802	0.4766	0.75	0.7899	0.7794	1.34

Table 3.12: Occurrence percentages of the proposed classes based on observed multi-year data and average of 100 generated synthetic sequences at three locations.

Class	SRRL BMS		UNLV		ORNL	
	Observed (%)	Synthetic (%)	Observed (%)	Synthetic (%)	Observed (%)	Synthetic (%)
1	18.3	14.7	45.2	34.9	12.9	6.5
2	0.9	6.8	0.2	6.2	0.1	6.8
3	2.1	1.5	0.6	0.8	5.8	4.5
4	29.4	33.6	29.6	36.4	20.8	20.4
5	21.9	16.7	9.3	8.0	17.6	21.2
6	5.6	3.6	1.6	1.0	15.1	13.7
7	8.7	13.9	7.3	9.9	5.8	9.8
8	12.6	7.0	6.0	2.1	19.9	10.3
9	0.1	1.7	0.1	0.3	1.4	6.3
10	0.3	0.4	0.1	0.3	0.6	0.3



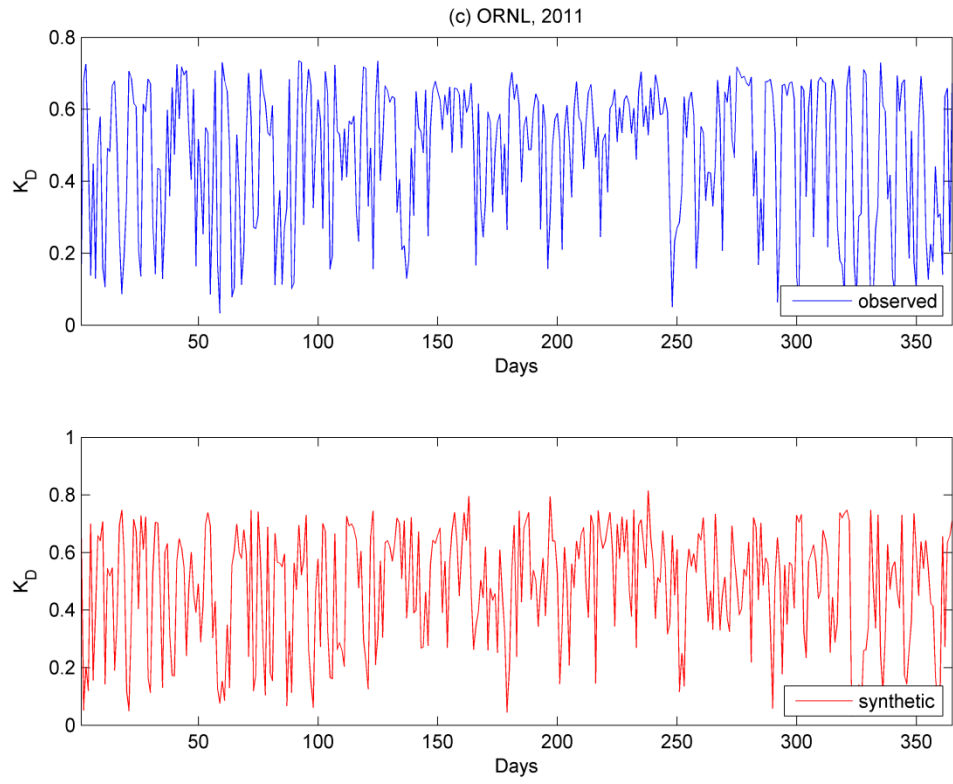
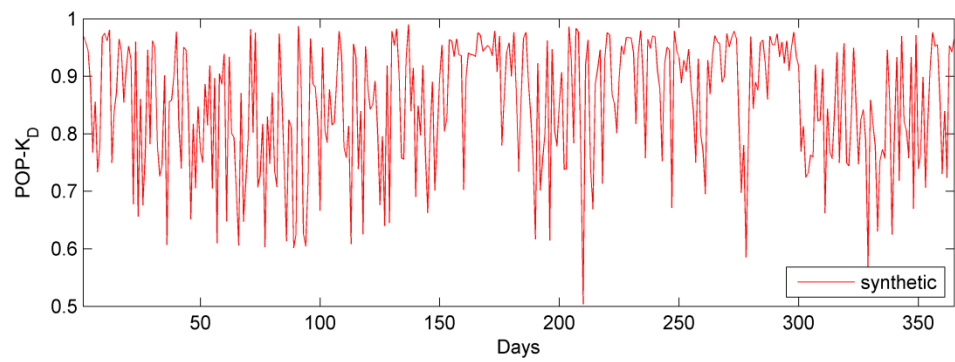
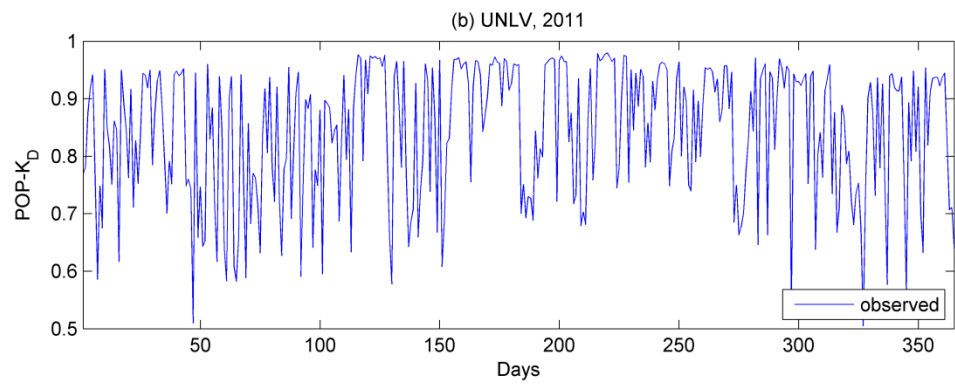
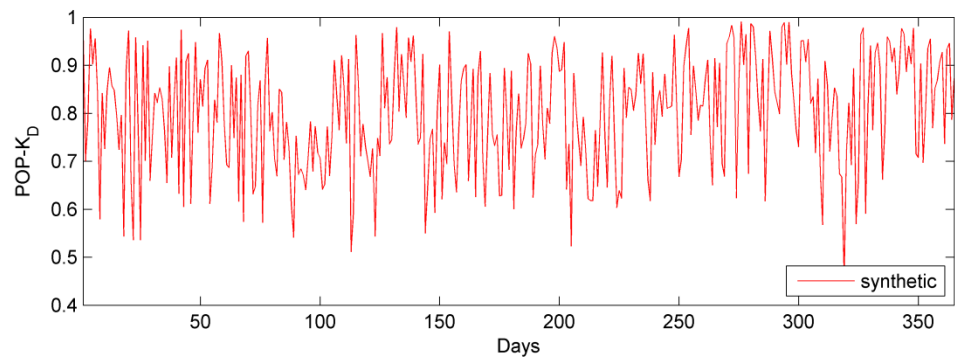
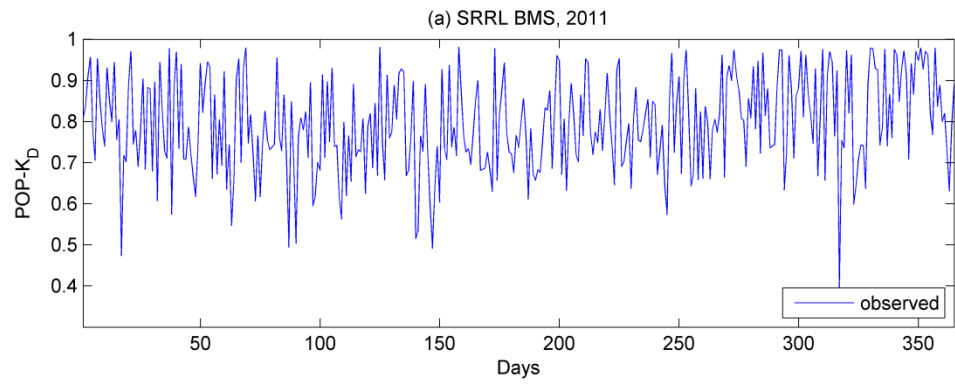


Fig. 3.14: Synthetic and observed sequences of K_D in 2011 for (a) SRRL BMS, (b) UNLV, and (c) ORNL.



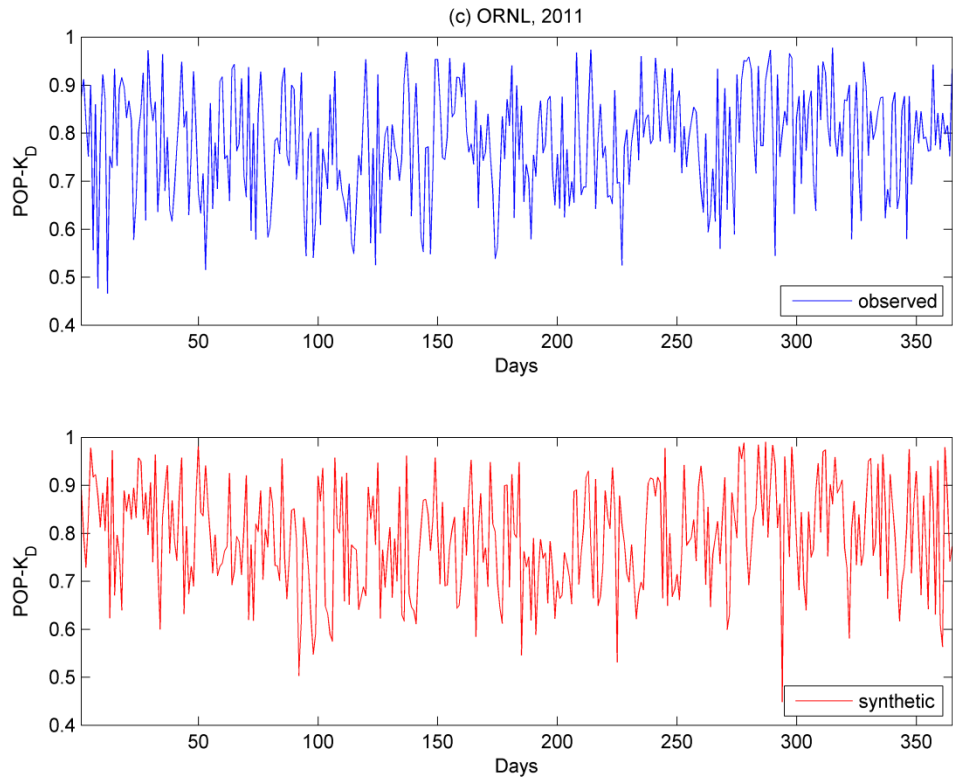


Fig. 3.15: Synthetic and observed sequences of POP-K_D in 2011 for (a) SRRL BMS, (b) UNLV, and (c) ORNL.

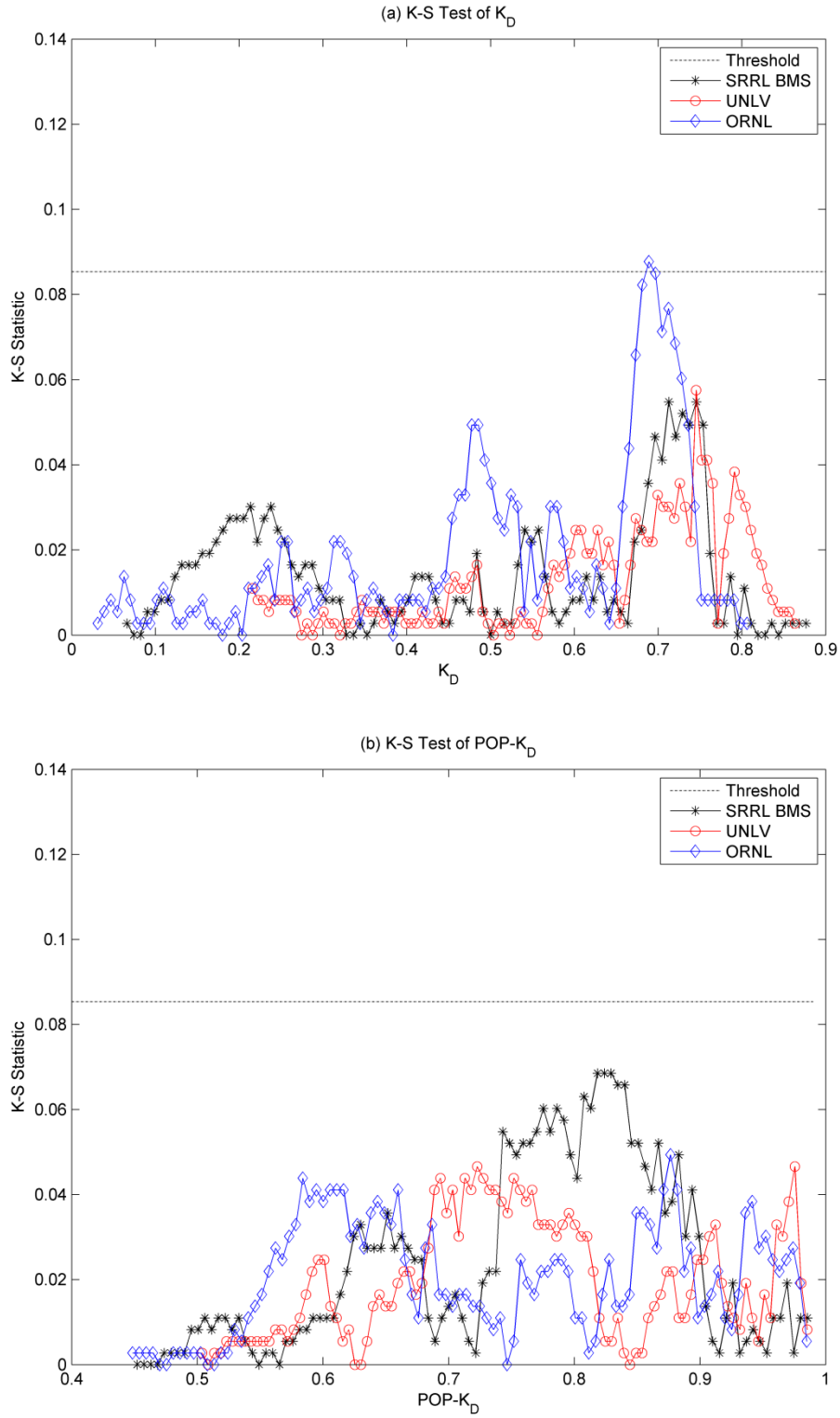


Fig. 3.16: K-S statistic observation of synthetic and observed sequences at three locations for (a) K_D and (b) POP- K_D .

3.6. Chapter contributions

This chapter proposes a new characterization and classification method for daily sky conditions by using K_D and POP- K_D . Quality of daily solar irradiance is characterized by a newly proposed parameter, POP- K_D . This characterized daily quality is varying and uncertain at the middle level of the quantity, but high and more certain at very high and low quantity levels. In addition, the proposed methods reveal interesting patterns. For multiple years, annual distributions of the K-POP method are enclosed by an irregular nonagon boundary for all three locations. This boundary has different specific ranges of POP- K_D at different K_D points, and consequently this shape of the boundary relates the quality of irradiance to the quantity. The annual mean of the 2-tuple (K_D and POP- K_D) at each location is statistically constant. Therefore, one annual statistical result may be representative of multiple-year statistical results at a certain location. In addition, the K-POP method relates the quantity of irradiance with the quality of irradiance through comparisons of seasonal trends of the 2-tuple at each location. The classification results show that several dominant classes exist commonly for all three locations, and transitions among the dominant classes are significantly frequent compared to transitions among other classes. Monthly patterns of the dominant classes at each location can be useful to characterize the local nature of solar irradiance.

This chapter also generates synthetic sequences of the 2-tuple based on a first-order Markov approach proposed by Aguiar et al. (1988). The synthetic sequences can be useful to optimize the solar-based generation system and substitute limited measured data. The Markov approach for each tuple consists of ten classes grouped by a monthly average of the tuple, and each class has a 10-by-10 transition matrix where each state is quantized by a maximum and minimum value among the allocated values in the class. The synthetic sequences are generated

for a year using the libraries of the matrices derived from the data at the three locations: SRRL BMS, UNLV, and ORNL. The generated sequences show statistical similarities with observed sequences.

CHAPTER 4. DAY-AHEAD QUANTITY ESTIMATION OF SOLAR IRRADIANCE USING SKY COVER FORECAST

4.1. Chapter introduction

This chapter proposes new methodologies providing accurate day-ahead solar irradiance quantity estimation using K_D based on the day-ahead NWS sky cover forecast. The proposed methods use data from the sky imager to revise the NWS sky cover forecast, and the revised data are subsequently used to generate solar irradiance estimation at the ground level. Data from the SRRL BMS had been used for model development and result verification at different stages.

The rest of this chapter is organized as follows. Section 4.2 presents the details about observed and forecast sky cover data and measured solar irradiance data in the selected station. Section 4.3 discusses the relationship between the NWS-based forecasted and the ground-based observed sky cover. Section 4.4 describes the conversion from cloud observation to solar irradiance. Section 4.5 proposes five estimation methods. Section 4.6 shows the test results of the proposed methods. Section 4.7 presents contributions of this chapter.

4.2. Data

For sky imager-based cloud information, ground-based observation of total sky cover for the SRRL BMS were retrieved from the Measurement and Instrumentation Data Center (MIDC) at the National Renewable Energy Laboratory (NREL) from 2009 through 2012 (MIDC, 2012). More details about the sky imager-based cloud data are provided in Section 4.3.

For cloud forecast information, historical total sky cover forecast data released from NWS were retrieved from the National Digital Forecast Database (NDFD) at the NOAA from 2009 through 2012 (NDFD, 2012). The NDFD archives the historical forecast data. The total sky cover forecast is updated with a grid resolution of 5 kilometers for the continental U.S. (NDFD, 2012). Thus, all retrieved data were specified to the closest grid point to SRRL BMS. Section 4.3 will provide further details about the retrieval process for the historical total sky cover forecast data from NDFD.

For K_D , I_{GHI} measured by the Kipp and Zonen CM22 pyranometer in 1-minute intervals over 4 years (2009–2012) at SRRL BMS were used for this study (MIDC, 2012). Data for I_{GHI} were integrated from sunrise to sunset for each day to calculate K_D .

4.3. Relationship between observed and forecast sky cover

4.3.1. Ground-based sky cover observation

Analysis of cloud images is useful for cloud observation in atmosphere. Automated image-processing schemes using ground-based sky imagers provide frequent cloud information with high resolution (Pfister et al., 2003; Long et al., 2006). For this study, the Yankee Model 880 Total Sky Imager (TSI) was used to capture images from the sky. The TSI automatically provides colored raw sky images of daytime sky conditions; then a software-based filter is applied to calculate the cloud coverage from the raw image (Long et al., 2006). This instrument differentiates the detected cloud fraction into cloudless, thin, and thick (opaque), and the total sky cover is the sum of thin and thick clouds (Pfister et al., 2003; Long et al., 2006). Sky cover images from 10° solar elevation angle after sunrise (80° zenith angle) to 10° solar elevation angle before sunset were collected in 1-minute intervals (MIDC, 2012). Total cloud observation is selected due to specific description of the cloud coverage over the entire atmosphere and close relation to surface solar irradiance. In addition, the forecast cloud information from NWS covers all types of clouds based on the ground-based cloud measure supplemented by remote sensing for clouds located above 12,000 feet (Yan and Zhao, 2009). For this study, the daily total sky cover (TKC_D) is defined by averaging the total daytime cloud coverage expressed in percentages.

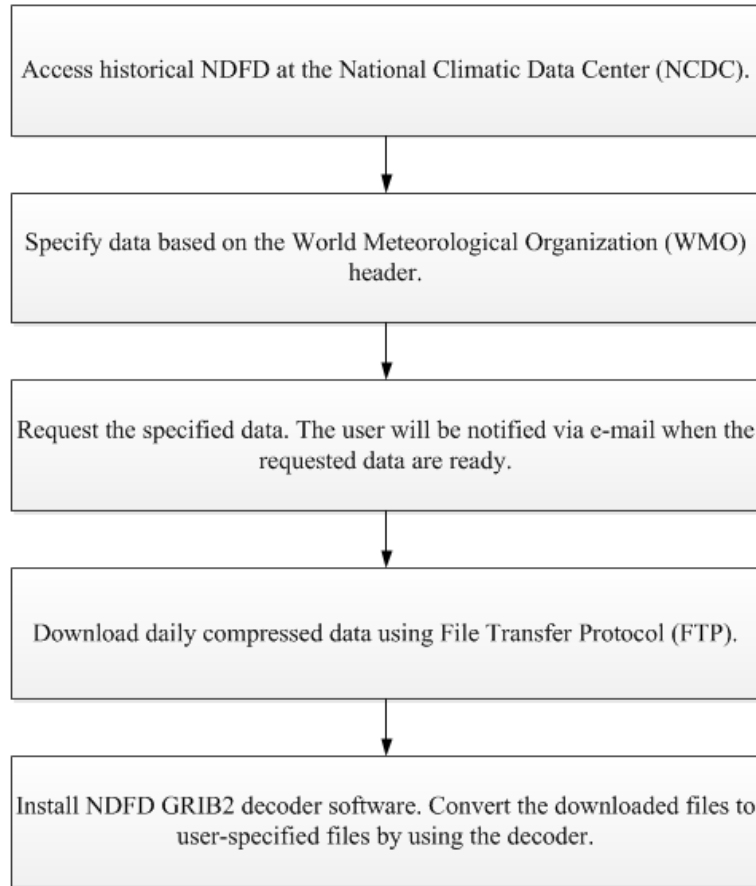
4.3.2. NWS day-ahead total sky cover

As mentioned in Chapter 2, the NWS forecast product is generated by the collaborative operation of national centers and regional offices using various measurements. The forecast sky cover is the two-digit percentage of cloud amount in the atmosphere valid for the indicated time (NWS, 2012). It is updated hourly to project every 3 hours out to 72 hours or every 6 hours out to 168 hours (NDFD, 2012).

Fig. 4.1a explains a retrieval process for the historical total sky cover data from NDFD. The data can be specified by using the Weather Meteorological Organization (WMO) header for the sky cover at a certain area; this step reduces data sizes for downloading. As all data provided by NDFD are gridded binary version 2 (GRIB2), decoder software is recommended to decode the GRIB2 files (NDFD, 2012).

The retrieved historical sky cover is a sequence projecting 3- or 6-hour intervals. This study selected the 3-hour interval for the shorter temporal resolution. Table 1 compares the absolute differences of the day-ahead NWS total sky cover values released at 18:00 and at 24:00 (midnight) for 10 days. Table 1 indicates that the sky cover values released at 18:00 are almost constant through midnight. This pattern is found for the entire year. Therefore, the sequence released at 18:00 was selected to provide earlier forecasting information to users. All times here are local to SRRL BMS, Mountain Standard Time (GMT -7).

(a) Retrieval of historical total sky cover forecasts from NDFD



(b) Sequence processing

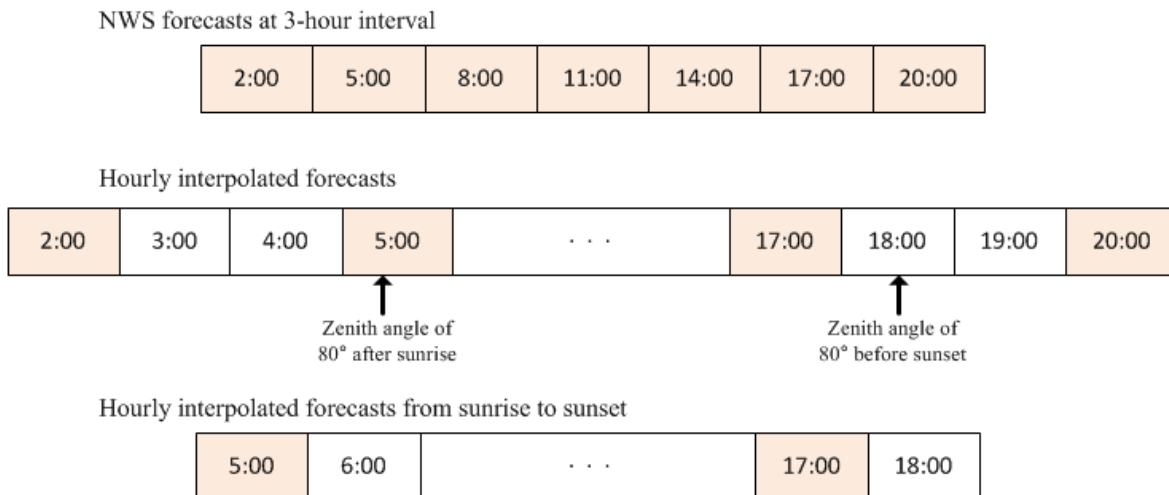


Fig. 4.1: Retrieval and processing of the NWS total sky cover; (a) retrieval of historical data from NDFD and (b) example of how the total sky cover forecasts are processed.

Table 4.1: A sample of the absolute differences between the next-day total sky cover forecast released by the National Weather Services at 18:00 and at midnight for SRRL BMS.

Day	Day-ahead target hour (Mountain Standard Time)			
	5:00	8:00	11:00	14:00
January 1, 2011	0	0	0	0
January 2, 2011	0	0	0	0
January 3, 2011	0	0	0	0
January 4, 2011	29%	6%	6%	0
January 5, 2011	6%	0	0	0
January 6, 2011	0	0	0	0
January 7, 2011	0	0	0	0
January 8, 2011	0	0	0	0
January 9, 2011	0	0	0	0
January 10, 2011	0	0	0	0

Each daily forecast sequence requires some processing for achieving the day-ahead average of the sky cover forecast, defined as $TKC_{D,NWS}$. Fig. 4.1b shows an example of the sequence processing. First, the 3-hour sequence is linearly interpolated into hourly increments. This step of converting from a 3- to a 1-hour sequence is also similarly conducted at local NWS offices for the forecast of sky coverage (Perry, 2012). Next, the interpolated sequence is filtered from the zenith angle of 80° after sunrise to 80° before sunset every day to correspond to the range used by the ground-based sky imager. Then, $TKC_{D,NWS}$ is calculated by taking the average of the filtered hourly forecast.

The historical NWS sky cover data for 3 years (2009–2011) were converted to $TKC_{D,NWS}$ using the sequence processing procedure. The $TKC_{D,NWS}$ data were then used in the next stage to develop their relationship with TKC_D . Sequence processing is also used to predict $TKC_{D,NWS}$ for estimating purpose. The predicted $TKC_{D,NWS}$ provides the overall estimation of day-ahead cloud quantity.

4.3.3. Relationship between $TKC_{D,NWS}$ and TKC_D

The relationship between $TKC_{D,NWS}$ and TKC_D is investigated to quantify the uncertainty of the NWS sky cover forecast. Both parameters range from 0% to 100%, where 0% indicates no cloud (i.e. clear sky), and 100% indicates overcast. These two parameters can be analyzed to investigate if there is any relationship between the forecast and the observation.

$TKC_{D,bias}$ is defined as the difference between the historical $TKC_{D,NWS}$ and the corresponding historical TKC_D . Fig. 4.2 shows the distributions of $TKC_{D,bias}$ at different solar irradiance levels (2009–2011). The quantity of daily solar irradiance can be classified by using K_D . High, medium, and low irradiance level can be defined for $K_D \geq 0.6$, $0.3 \leq K_D < 0.6$, and $K_D < 0.3$, respectively, as shown in Chapter 3. $K_D = 0.3$ and 0.6 correspond to $TKC_D = 0.8559$ and 0.4106 , respectively, based on Eq. (4.3) in Section 4.4.

Table 4.2 shows the statistical parameters, mean (μ) and standard deviation (σ), of the distributions shown in Fig. 4.2. These parameters are estimated by the assumption that each distribution follows a normal distribution as $TKC_{D,bias} \sim N(\mu, \sigma^2)$. The parameters in Fig. 4.2d are derived with the assumption of a truncated normal distribution, applied when the normal distribution is restricted within a finite range (Robert, 1995). Figs. 4.2a, 4.2b, and 4.2c are

distributed in $[-1, 1]$, but Fig. 4.2d is limited as $[-1, 0.1414]$. $TKC_{D,bias}$ in Fig. 4.2d cannot exceed 0.1441 because the maximum value of $TKC_{D,NWS}$ is 1 and TKC_D is limited to 0.8559.

Table 4.2 also includes the proportion of the three different K_D levels in terms of the number of days in the 3-year period. Totally observation data for 1075 days were obtained. The high K_D level is the most dominant with approximately 60%. As shown in Table 4.2, the forecast generally tends to be overestimated because μ of $TKC_{D,bias}$ for the overall 3-year observation is 0.1148. This positive value is mostly caused by the strong overestimation at the high K_D level ($=0.1986$). The overestimation significantly decreases at the low K_D level and nearly disappears at the medium K_D level.

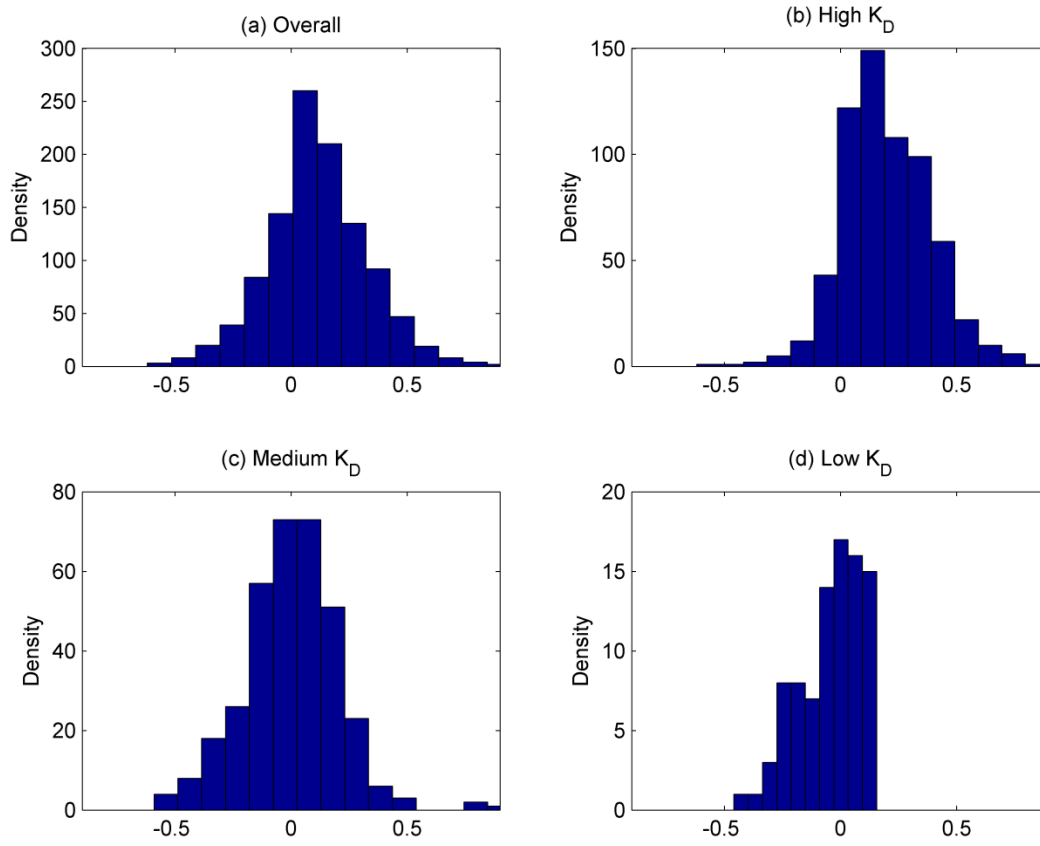


Fig. 4.2: Distribution of $TKC_{D,bias}$ for (a) overall 3-year observation and for days with (b) high K_D , (c) medium K_D , and (d) low K_D levels.

Table 4.2: Mean and standard deviation of $TKC_{D,bias}$ at different K_D levels for three years (2009 – 2011).

	Overall	High K_D	Medium K_D	Low K_D
Number of days	1075	642 (59.72%)	352 (32.74%)	81 (7.53%)
Mean (μ)	0.1148	0.1986	0.0008	0.0513
Std. deviation (σ)	0.2174	0.1915	0.2047	0.1943

4.3.4. Verification

The accuracy of $\text{TKC}_{\text{D,for}}$ (the estimated TKC_{D}) is verified by computing $\Delta\text{TKC}_{\text{D}}$, defined as the difference between $\text{TKC}_{\text{D,for}}$ and the corresponding TKC_{D} obtained from observation. Data were obtained for the year 2012 at SRRL BMS for this purpose (data for 345 days; data for 21 days were missing). Table 4.3 summarizes μ and σ of the distributions of $\Delta\text{TKC}_{\text{D}}$ at different irradiance levels. $\text{TKC}_{\text{D,for}}$ is calculated by the five proposed quantity methods which use different adjustment schemes for $\text{TKC}_{\text{D,bias}}$. More details about these methods will be described in Section 4.5.

From Table 4.3, Methods 1 and 5, which use zero adjustment, show the largest μ of $\Delta\text{TKC}_{\text{D}}$ at the high K_{D} level. However, Methods 2, 3, and 4, which use different adjustment schemes, show significantly smaller μ of $\Delta\text{TKC}_{\text{D}}$ at this level. This comparison indicates that appropriate adjustment schemes can reduce $\Delta\text{TKC}_{\text{D}}$. The performance at the high irradiance level is the most important since this level is dominant, approximately 60%, as indicated in Table 4.2. In addition, Methods 3 and 4, which use the conditional adjustment for different K_{D} levels, result in smaller differences for medium and low levels when compared to Method 2 which uses the uniform adjustment for all levels. This comparison confirms that appropriately refined adjustment is even more effective.

As shown in Table 4.3, all five methods show relatively constant σ for $\Delta\text{TKC}_{\text{D}}$. This characteristic is similar to the pattern of σ for $\text{TKC}_{\text{D,bias}}$ shown in Table 4.2. As shown later, this result is useful for the uncertainty analysis for the proposed quantity methods.

Table 4.3: Mean and standard deviation of ΔTKC_D at different K_D levels for one year (2012).

Method	Overall		High K_D		Medium K_D		Low K_D	
	μ	σ	μ	σ	μ	σ	μ	σ
1 and 5	0.0315	0.2356	0.1227	0.1930	-0.1137	0.2299	-0.1269	0.1944
2	-0.0801	0.2349	0.0126	0.1911	-0.2275	0.2265	-0.2417	0.1944
3	-0.0603	0.2614	0.0032	0.2349	-0.1653	0.2847	-0.1494	0.1775
4	-0.0268	0.2475	0.0474	0.2162	-0.1460	0.2606	-0.1494	0.1775

Observation for the relationship between $TKC_{D,NWS}$ and TKC_D verifies the NWS sky cover forecast based on the type of day (high, medium, and low irradiance). Such information is difficult to be obtained in the previously reported direct conversion from the sky cover forecast to solar irradiance. The verification result shows about 10% overestimated overall bias at the selected location. The bias is increased up to about 20% at days with high irradiance, but approximately disappeared at days with medium and low irradiance. Thus, adaptive bias adjustment at different daily irradiance levels is recommended in the estimation method.

Verification of cloud forecast has been performed mostly by using satellite-based cloud observation (Sohne et al., 2008; Shah et al., 2010; Ye and Chen 2013). This study used the sky imager-based cloud observation for the verification. The sky imager can be a substitute for the satellite because the ground-processed images provide accurate cloud tracking results (Kazantzidis et al., 2012; Escrig et al., 2013).

4.4. Relationship between observed sky cover and ground-based solar irradiance

4.4.1. Relationship between TKC_D and K_D

The relationship between TKC_D and K_D is investigated to observe the uncertainty for the cloud-to-irradiance conversion and to establish an accurate conversion curve. Negative correlation had been reported for this relationship (Pfister et al., 2003).

Fig. 4.3 shows this relationship based on the 3-year data at SRRL BMS (2009–2011). The relationship shown in Fig. 4.3 is not linear. Furthermore, K_D generally shows narrower ranges of dispersion at low TKC_D values (clearer sky) but wider ranges at high TKC_D values (more cloudy sky). These different uncertainty patterns make it difficult to develop the conversion equation between TKC_D and K_D with uncertainty quantified. For this reason, more sophisticated fitting technique has been selected over a polynomial-based fitting approach. The Box-Cox transformation technique was chosen to linearize the relationship between TKC_D and K_D .

The Box-Cox transform converts a positive definite of the dependent variable y with a function:

$$y_i^{(\lambda)} = \begin{cases} \frac{y_i^\lambda - 1}{\lambda} & \text{when } \lambda \neq 0 \\ \log(y_i) & \text{when } \lambda = 0 \end{cases} \quad (4.1)$$

where λ is the power parameter (Box and Cox, 1964). Fig. 4.4 shows the relationship between the original TKC_D and the transformed K_D ($K_{D,\text{trans}}$) with $\lambda = 2.09$. The pattern in Fig. 4.4 shows that the relationship is approximately linear with evenly distributed uncertainty for all ranges.

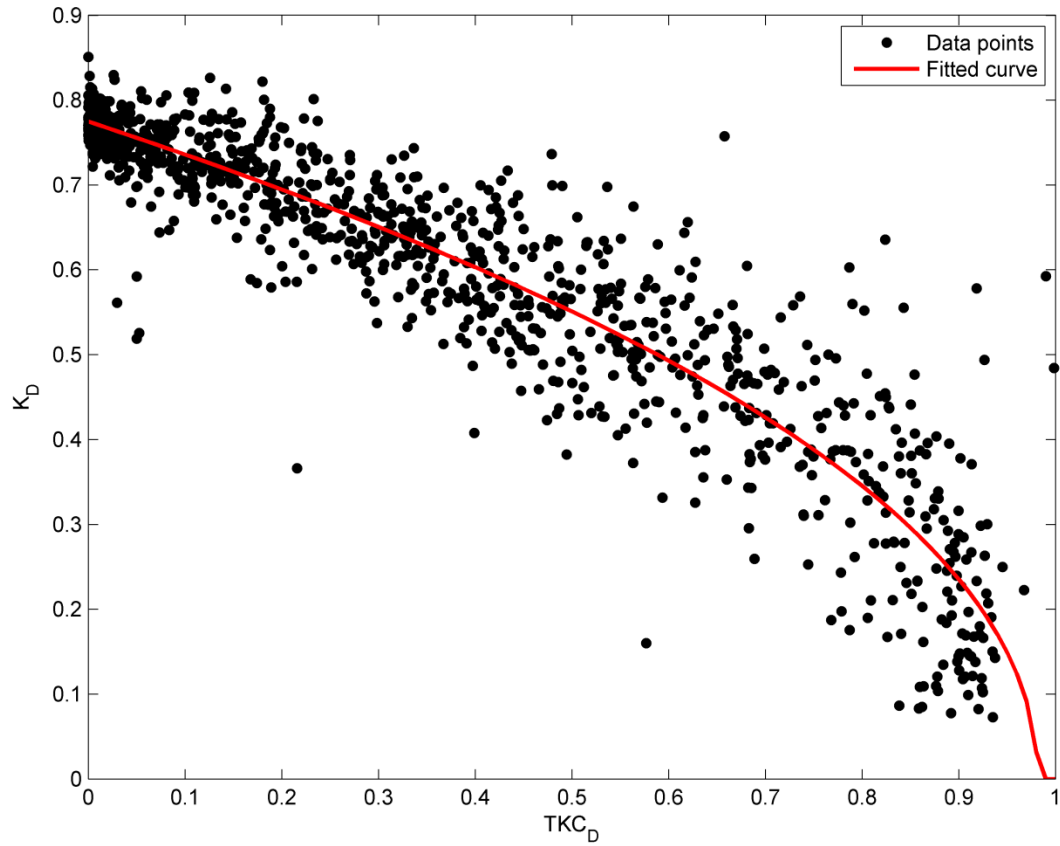


Fig. 4.3: Relationship between observed TKC_D and measured K_D .

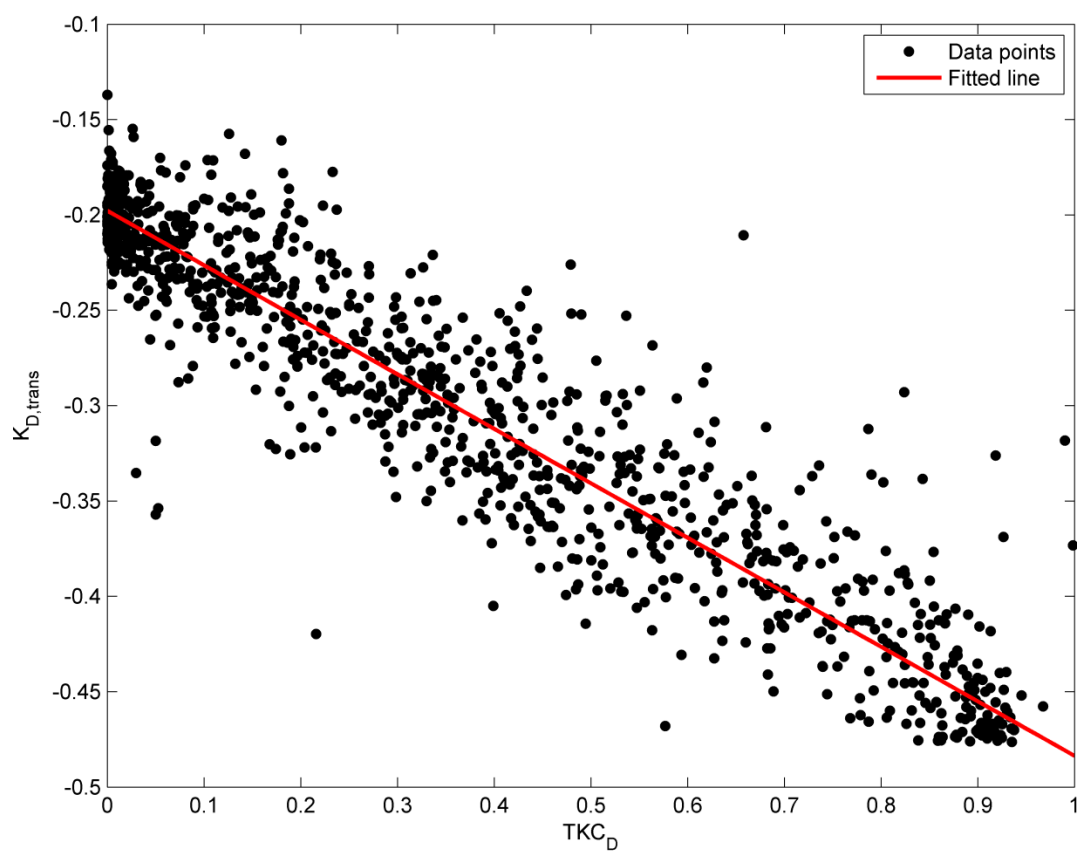


Fig. 4.4: Relationship between observed TKC_D and Box-Cox transformed K_D .

Bisquare robustness is a robust regression technique that uses Tukey's biweight function for increasing resistance to a wild data point (Rousseeuw and Leroy, 1987). Using least square fitting with bisquare weights, the pattern shown in Fig. 4.4 can be described by the following linear equation:

$$\begin{aligned}
 K_{D,trans} &= a_1 TKC_D + b_1 \\
 a_1 &= -0.2825 [-0.2882, -0.2768] \\
 b_1 &= -0.1979 [-0.2006, -0.1953]
 \end{aligned} \tag{4.2}$$

where the estimated coefficients a_1 and b_1 are specified with their respective 95% CI. The fitted line in Fig. 4.4 can be inverse-transformed back to the form shown in Fig. 4.3 by the following equation:

$$K_D = (0.5864 - 0.5904 TKC_D)^{0.4785}. \tag{4.3}$$

Eq. (4.3) can be used to estimate the amount of daily solar irradiance at ground level (in terms of K_D) using the forecast sky cover information.

4.4.2. Verification

The accuracy of the TKC_D -vs- K_D curve was verified by the comparison of observed K_D and K_D obtained from observed TKC_D by using the conversion curve. The data for the year 2012 were used for this purpose. ΔK_D is defined as the difference between the observed and obtained K_D . Table 4.4 shows the statistical parameters of the distributions of ΔK_D for the entire year and for days with high, medium, and low K_D levels. These results show that overall ΔK_D are small and the usefulness of Eq. (4.3) in converting TKC_D to K_D was demonstrated.

The values of ΔK_D shown in Table 4.4 are significantly lower than those for ΔTKC_D in Table 4.3. This indicates that the main source of uncertainty for the proposed quantity estimation method is identified as the NWS forecast uncertainty ($\text{TKC}_{D,\text{bias}}$) rather than the uncertainty for the TKC_D -vs- K_D conversion process.

Table 4.4: Mean and standard deviation of ΔK_D at different K_D levels for one year (2012).

	Overall	High K_D	Medium K_D	Low K_D
Number of days	345	213 (61.74%)	111 (32.17%)	21 (6.09%)
Mean (μ)	-0.0155	-0.0280	-0.0091	0.0767
Std. deviation (σ)	0.0725	0.0612	0.0782	0.0791

Eq. (4.3) is a newly proposed TKC_D -vs- K_D curve. Compared to previous reported results from the viewpoint of estimating the amount of received irradiance based on cloud information, this curve provides simplicity with accuracy.

Kasten and Czeplak (1979) and Perez et al. (2007) proposed equations that use a ratio between I_{GHI} and the clear sky irradiance. The clear sky irradiance is varying at different spatial and instrumental conditions and needs to be established for different locations using different instruments. In addition, the conversion equation (Perez et al., 2007) had been developed with relatively high uncertainty levels that have not yet been quantified.

Comparatively, the proposed curve uses the sky clearness index, a ratio between I_{GHI} and the extraterrestrial irradiance. The extraterrestrial irradiance can be readily calculated if geographical information and time of the year/day are provided. In addition, the proposed curve improves the conversion accuracy because the applied linearization technique (Box-Cox transform) provides more accurate fitting result. Furthermore, the uncertainty of the proposed curve can be quantified with 95% CI as discussed in Section 4.5.

4.5. Proposed estimation methods for day-ahead K_D

This chapter proposes five day-ahead K_D estimation methods that are variations of the 3-stage sequence.

- Stage 1: Processing the NWS day-ahead total sky cover forecasts into $TKC_{D,NWS}$ data as described in Section 4.3.2.
- Stage 2A: Adjusting the $TKC_{D,NWS}$ data into $TKC_{D,for}$ with μ_{adj} by the following equation:

$$TKC_{D,for} = TKC_{D,NWS} - \mu_{adj}. \quad (4.4)$$

- Stage 3B: Estimating $K_{D,for}$ from $TKC_{D,for}$ based on Eq. (4.3).

4.5.1. Proposed Method 1

As described in Fig. 4.5, Method 1 uses zero for μ_{adj} in Eq. (4.4), and then converts this value into $K_{D,for}$. This is similar to the direct cloud-to-irradiance conversion. The direct conversion was proposed by the previous studies (Perez et al., 2007; Perez et al., 2010) for estimating hourly solar irradiance, but this study modified for estimating daily solar irradiance. Compared to the other proposed quantity methods, this method has the benefit of simplicity.

The 95% CI of $K_{D,for}$ for this method is estimated as follows. The uncertainty of a single day-ahead cloud prediction in the NWP model can be assumed as approximately 15% to 16% (Shah et al., 2010). Tables 4.2 and 4.3 show that σ (measure of uncertainty) of both $TKC_{D,bias}$ and ΔTKC_D is approximately 0.2 for all levels of K_D . Thus, the uncertainty at the output of Stage 2A may be approximated as 0.2, with most of it coming from the NWS forecast.

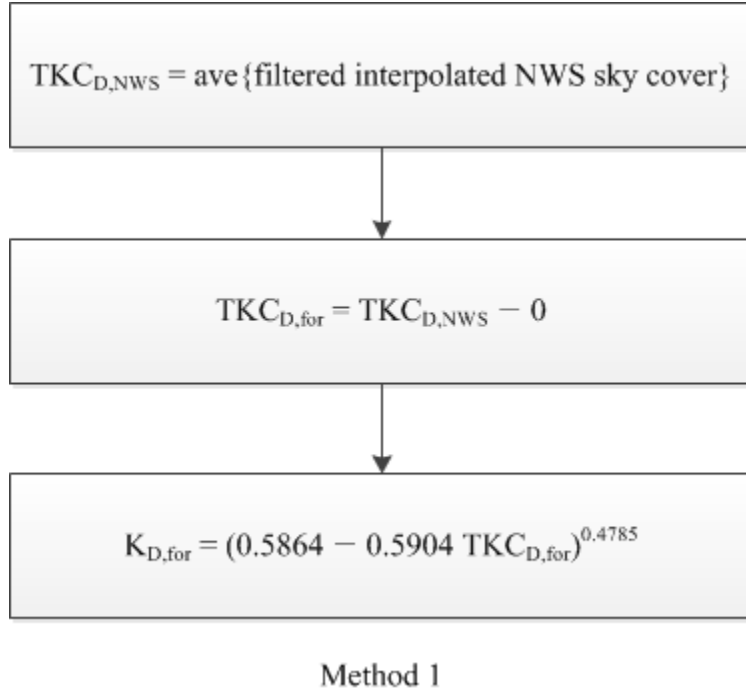


Fig. 4.5: Algorithms for the proposed Method 1.

For the uncertainty introduced in Stage 3A, the standard deviation of $K_{D,trans}$ (σ_{3A}) computed by using Eq. (4.2) is determined by the following equation:

$$\sigma_{3A} = \sqrt{(c_{a1}\sigma_{a1})^2 + (c_{b1}\sigma_{b1})^2 + (c_{2A}\sigma_{2A})^2} \quad (4.5)$$

where c_{a1} , c_{b1} , and c_{2A} are sensitivity coefficients and σ_{a1} , σ_{b1} , and σ_{2A} are standard deviations of a_1 , b_1 , and $TKC_{D,for}$, respectively (Ellison and Williams, 2012). The sensitivity coefficients are determined by taking the partial differential of Eq. (4.2) with respect to the respective parameters as following:

$$c_{a1} = \partial K_{D,trans} / \partial a_1 = TKC_{D,for}, \quad (4.6)$$

$$c_{b1} = \partial K_{D,trans} / \partial b_1 = 1, \quad (4.7)$$

$$c_{2A} = \partial K_{D,trans} / \partial TKC_{D,for} = a_1 = -0.2825. \quad (4.8)$$

Based on the CIs of a_1 and b_1 , σ_{a1} and σ_{b1} are calculated as:

$$\sigma_{a1} = |-0.2768 - (-0.2882)| / (2 \cdot 1.96) = 0.002908, \quad (4.9)$$

$$\sigma_{b1} = |-0.1953 - (-0.2006)| / (2 \cdot 1.96) = 0.001327. \quad (4.10)$$

Only c_{a1} depends on $TKC_{D,for}$ but σ_{a1}^2 is a very small number. Thus, σ_{3A} has an approximately constant value of 0.0565 regardless of $TKC_{D,for}$. Finally, the upper and lower boundaries of the 95% CI of $K_{D,for}$ are obtained by adding $\pm 1.96\sigma_{3A}$ into Eq. (4.2) and then inverse-transformed as:

$$Upper(K_{D,for}) = (0.8178 - 0.5904 TKC_{D,for})^{0.4785}, \quad (4.11)$$

$$Lower(K_{D,for}) = (0.3549 - 0.5904 TKC_{D,for})^{0.4785}. \quad (4.12)$$

Fig. 4.6a describes these boundaries along with the TKC_D -vs- K_D curve. Although σ_{3A} is constant for all $TKC_{D,for}$ values, the inverse-transformed CIs using Eqs. (4.11) and (4.12) show narrower ranges of dispersion at low $TKC_{D,for}$ values but wider ranges at high $TKC_{D,for}$ values. This characteristic of the calculated CIs reflects well the relationship between the observed TKC_D and measured K_D as shown in Fig. 4.3.

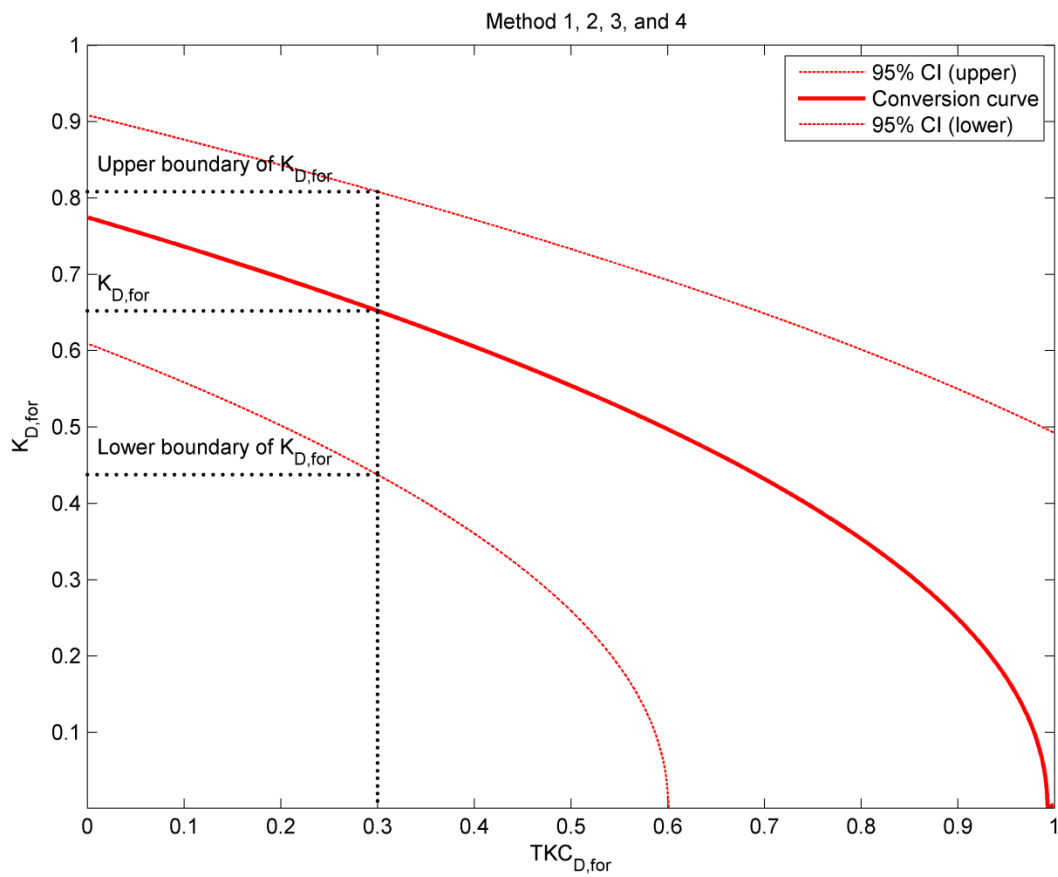
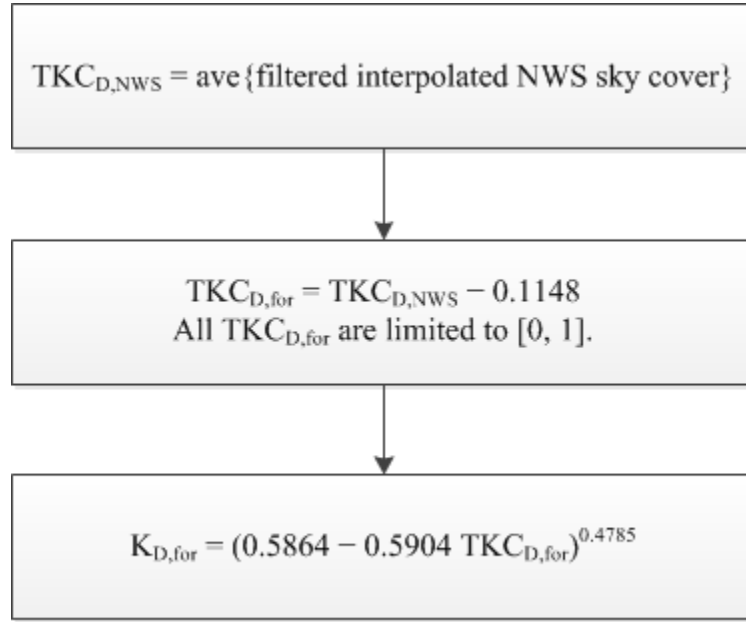


Fig. 4.6: Conversion curves and the boundaries for the 95% confidence interval (CI) for the proposed Methods 1, 2, 3, and 4.

4.5.2. Proposed Method 2

Method 2 uses the overall 3-year mean (from Table 4.2) of 0.1148 for μ_{adj} . Fig. 4.7 describes the steps of this method. The rationale for this method is a simple adjustment for $TKC_{D,bias}$ based on multi-year observation. The CI calculation for this method is the same as Method 1 as the uncertainty for the multi-year average remains at approximately 0.2. Fig. 4.6 also shows the boundaries of the CIs for this method.



Method 2

Fig. 4.7: Algorithms for the proposed Method 2.

4.5.3. Proposed Method 3

Method 3 conditionally adjusts μ_{adj} at different K_D levels as describe in Fig. 4.8. Based on Table 4.2, μ_{adj} is selected to be 0.1986, 0.0008, and 0.0513 at the high, medium, and low K_D level, respectively. The rationale for this method is to use more refined bias adjustment at different K_D levels based on historical observation data. This method also has the same CI as Method 1 because the uncertainty (represented by the standard deviation) for the three multi-year average is approximately 0.2 for all (Table 4.2).

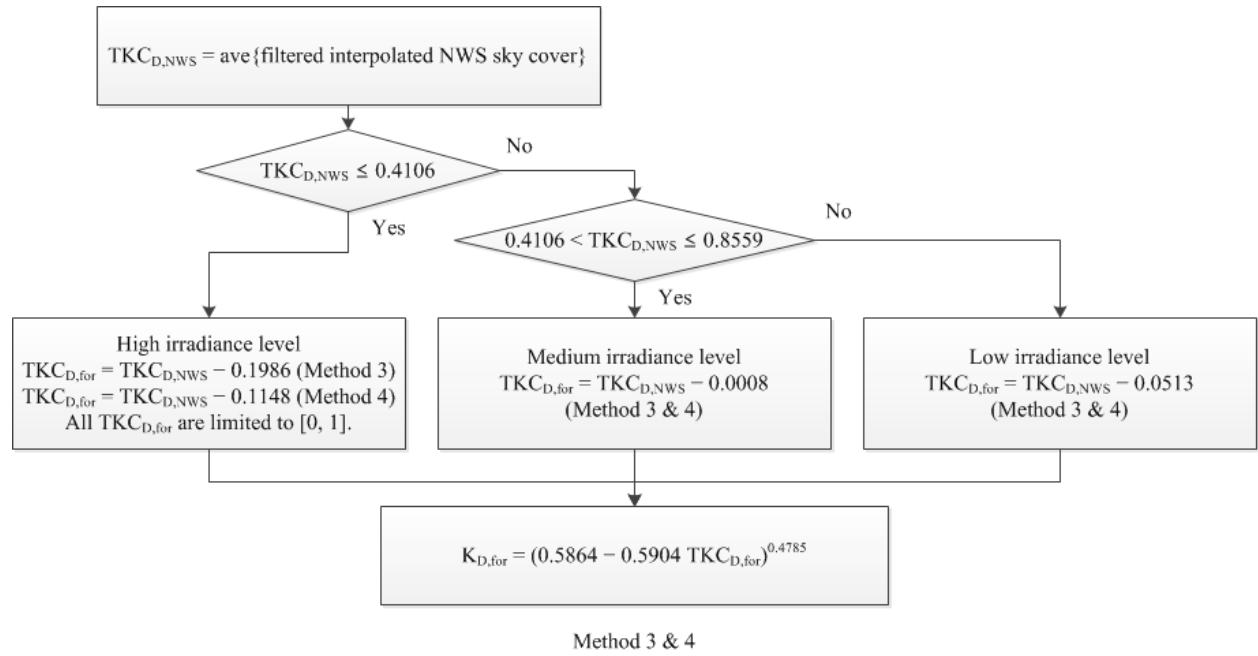


Fig. 4.8: Algorithms for the proposed Methods 3 and 4.

4.5.4. Proposed Method 4

Method 4 is similar to Method 3, but it uses the overall 3-year mean of 0.1148, as in Method 2, instead of 0.1986 for only the high K_D level (Fig. 4.8). The rationale for this method is that, based on the experience of using Methods 2 and 3, the multi-year overall average provides better estimation results for the high K_D level but not for the medium and low K_D levels. Since this method may be considered a hybrid of Method 2 and Method 3, the CI of this method is the same as the CI of Method 1 for the same reason.

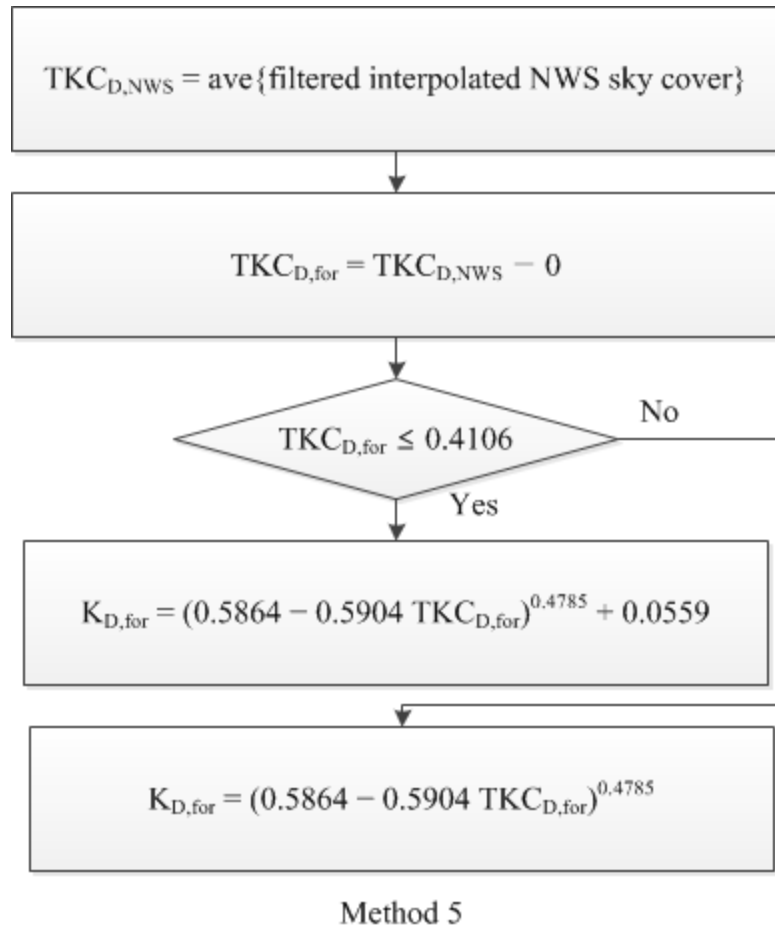


Fig. 4.9: Algorithms for the proposed Method 5.

4.5.5. Proposed Method 5

Method 5 implements the effect of the bias by shifting the pertinent portion of the original TKC_D -vs- K_D curve so that the bias has an impact in Stage 3A. The shift is only applied for the high K_D level where the bias is consistently more significant (as shown in Table 4.2). As shown in Fig. 4.9, the curve shifts up uniformly by an empirically determined amount for TKC_D below 0.4106. The first rationale for this method is to eliminate the gap in the adjustment process using Eq. (4.4). Fig. 4.10 shows the gap generated in Method 4 at the high K_D level ($TKC_D \leq 0.4106$). Method 4 uses 0.1148 for μ_{adj} at this level, and thus it always causes a gap between 0.2958 (subtracting 0.1148 from 0.4106) and 0.4106. Method 5 resolves this issue by moving the curve up and maintains continuity of the TKC_D values on the x-axis (TKC_D). The second rationale for this method is to utilize more fully the range of $K_{D,for}$. The y-intercept of the original curve is 0.7746 as shown in Fig. 4.10. The y-intercept indicates $K_{D,for}$ with no cloud ($TKC_D = 0$). However, K_D values larger than 0.7746 are observed (Fig. 4.3). With these two rationales in mind, the amount of shift is empirically determined to be 0.0559. The shift-up in this method increases the y-intercept to 0.8305 ($= 0.7746 + 0.0559$), as shown in Fig. 4.10, and consequently $K_{D,for}$ can be estimated to this value on average. The 95% CIs for this method are also shifted up along with the conversion curve as following:

$$Upper(K_{D,for}) = (0.8178 - 0.5904 TKC_{D,for})^{0.4785} + 0.0559 \quad \text{when} \quad TKC_{D,for} \leq 0.4106, \quad (4.13)$$

$$Lower(K_{D,for}) = (0.3549 - 0.5904 TKC_{D,for})^{0.4785} - 0.0559 \quad \text{when} \quad TKC_{D,for} \leq 0.4106. \quad (4.14)$$

Fig. 4.11 shows the shifted boundaries for this method.

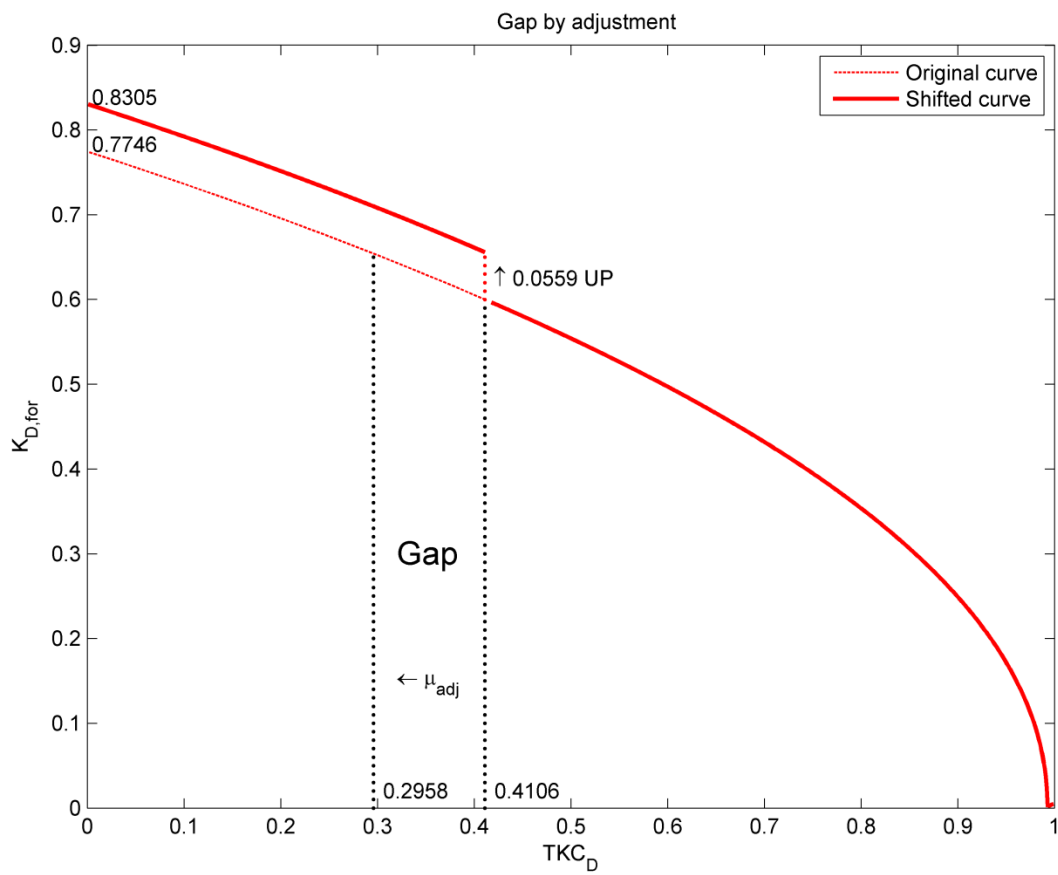


Fig. 4.10: Gap generated by adjustment in Stage 2A.

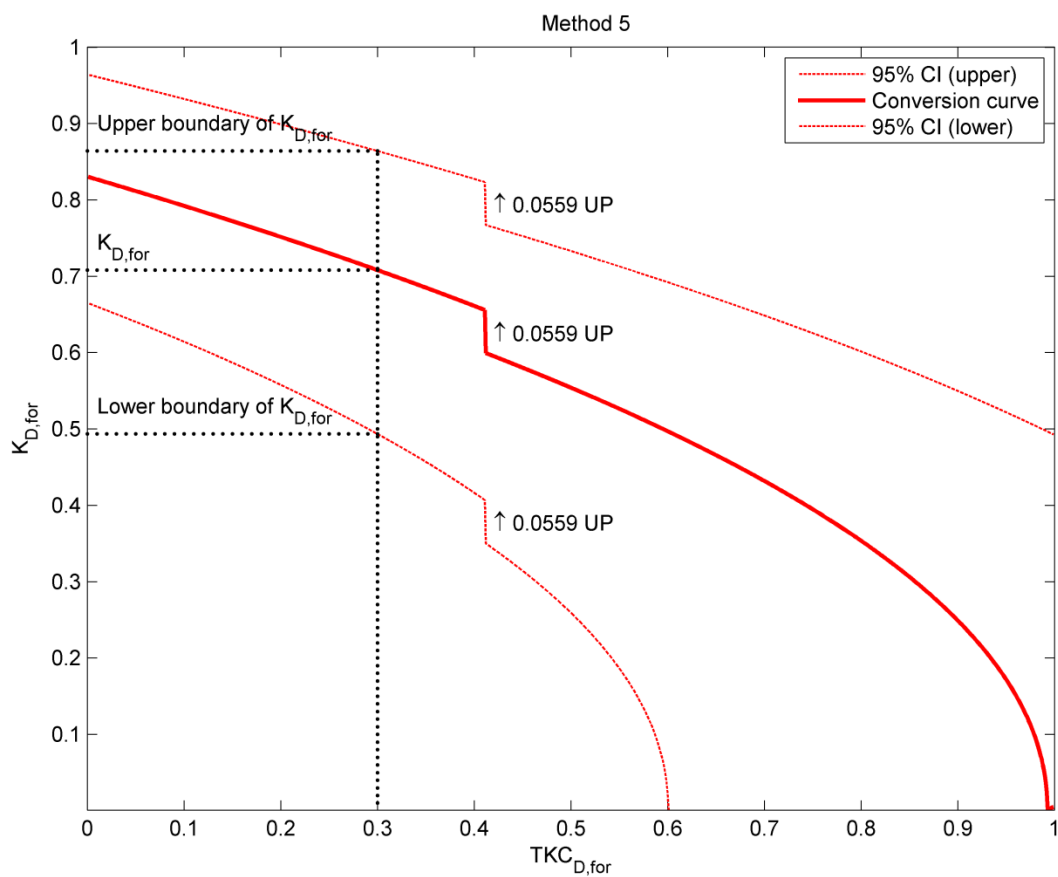


Fig. 4.11: Conversion curve and the 95% CI boundaries for Method 5.

4.6. Results

All five proposed quantity estimation methods were applied to predict $K_{D,for}$ by using the $TKC_{D,NWS}$ data for the year 2012 at SRRL BMS. The predicted $K_{D,for}$ values were compared to the corresponding values of the observed K_D in 2012 for verification.

Table 4.5 shows the Root-Mean-Square Error (RMSE) for the five proposed methods at different K_D levels. All five methods perform best at the high K_D level. Out of the five methods, Method 1 shows the highest RMSE at the high K_D level. Method 2 shows the most improved performance at the high K_D level, but its performances at the medium and low levels are strongly degraded. Performance of Methods 3, 4 and 5 generally show similar patterns.

Table 4.5: Comparison of RMSEs for the proposed quantity estimation methods.

Method	Overall		High K_D		Medium K_D		Low K_D	
	RMSE	%	RMSE	%	RMSE	%	RMSE	%
1	0.1286	20.56	0.1089	15.17	0.1338	28.55	0.2382	114.76
2	0.1312	20.98	0.0737	10.27	0.1628	34.73	0.2963	142.79
3	0.1331	21.28	0.0983	13.71	0.1596	34.06	0.2414	116.33
4	0.1285	20.55	0.0980	13.66	0.1480	31.58	0.2414	116.33
5	0.1308	20.92	0.0986	13.75	0.1543	32.93	0.2382	114.76

The Mean-Biased Errors (MBEs) for the five methods are summarized in Table 4.6. Similar to the RMSE result, most methods show better performance at the high K_D level. Method 2 is the best at high K_D but worst at medium and low K_D . Methods 4 and 5 show the smallest MBE for overall compared to others. In addition, all methods except Method 1 show

overestimation for the high K_D level but underestimation for the medium and low K_D levels. These patterns are very similar to those of the mean (μ) of ΔTKC_D in Table 4.3. This indicates that the MBE of the proposed methods strongly depend on their performance at Stage 2A.

The five methods were tested by using these parameters with a 99% confidence level and the results are summarized in Table 4.7. Out of the five methods, Method 5 has the smallest KSI and OVER.

Table 4.6: Comparison of MBEs for the proposed quantity estimation methods.

Method	Overall		High K_D		Medium K_D		Low K_D	
	MBE	%	MBE	%	MBE	%	MBE	%
1	-0.022	-3.66	0.081	11.32	-0.056	-12.17	-0.163	-84.19
2	0.036	6.02	0.031	4.31	-0.122	-26.33	-0.264	-136.92
3	0.018	3.05	0.031	4.38	-0.08	-17.35	-0.196	-101.36
4	0.005	0.80	0.049	6.88	-0.072	-15.65	-0.196	-101.36
5	0.007	1.11	0.042	5.89	-0.071	-15.34	-0.163	-84.19

Table 4.7: Comparison of KSI and OVER values for the proposed quantity estimation methods.

Method	KSI (%)	OVER (%)
1	65.93	21.33
2	74.81	12.87
3	44.62	0.96
4	41.85	0.27
5	29.40	0.25

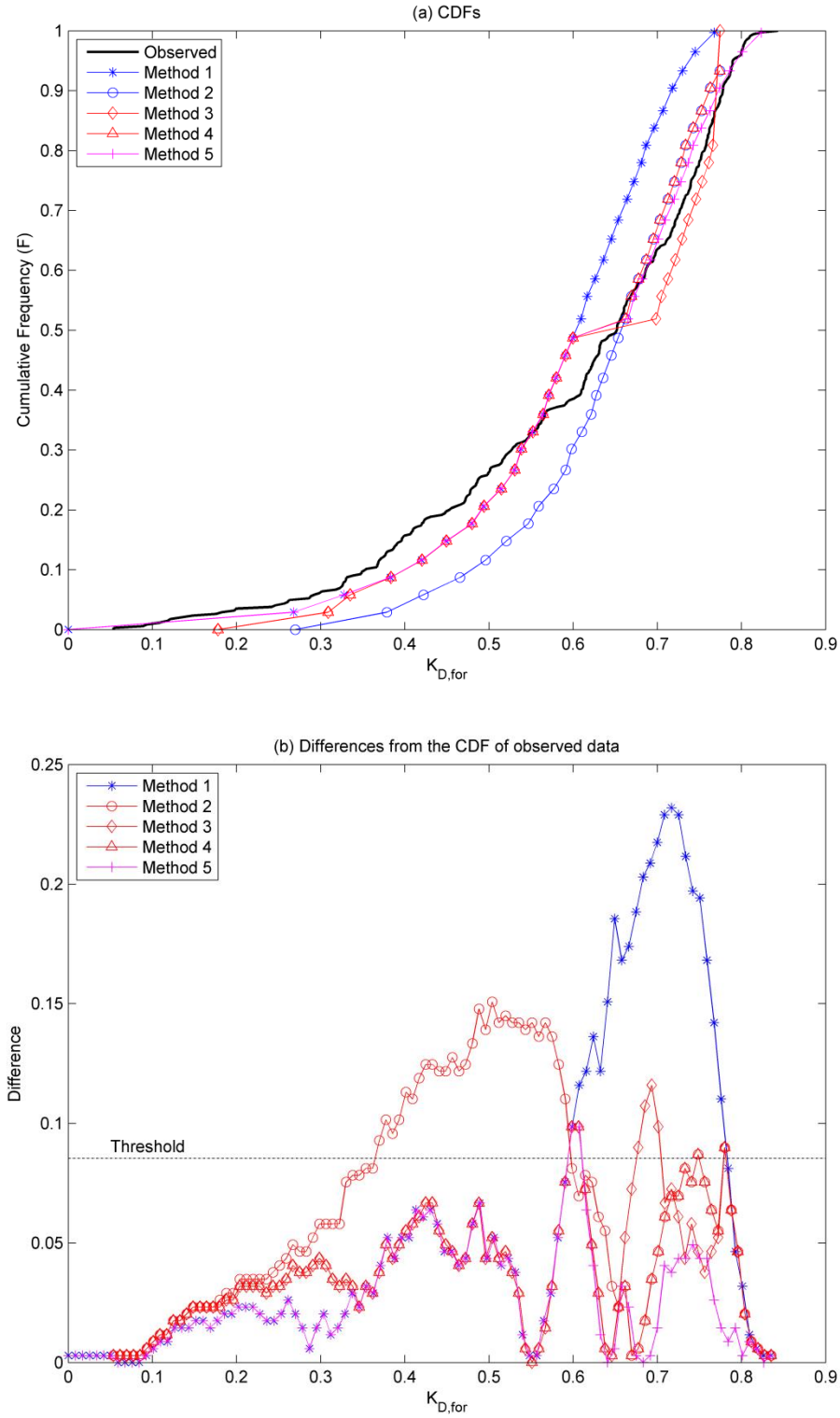


Fig. 4.12: K–S statistics for the five proposed quantity estimation methods: (a) CDFs and (b) differences of CDFs generated by the proposed methods and the CDF of the observed data.

Fig. 4.12a depicts the CDFs of the estimation results from using the five methods as well as the CDF of the observation data. Fig. 4.12b shows the differences between the CDFs of the five methods and the CDF of observed data over all ranges of $K_{D,for}$. As shown in Fig. 4.12b, Methods 3, 4, and 5 mostly stay under the threshold while Method 1 is significantly above the threshold at high level ($K_{D,for} > 0.6$) and Method 2 at medium level ($0.3 < K_{D,for} < 0.6$).

The above statistical results and the CDF observations indicate the need for applying $TKC_{D,bias}$ in the adjustment step in either Stage 2A (Methods 2, 3, and 4) or Stage 3A (Method 5). Method 1 which uses zero adjustment shows the highest error at the high K_D level in terms of RMSE and MBE. In addition, the CDF differences of Method 1 described in Fig. 4.12b are conspicuously over the threshold in high irradiance range, and thus Method 1 shows the highest OVER in Table 4.7. The estimation performance at high irradiance level is important for this location because days with high irradiance happen more than 60%. Therefore, $TKC_{D,bias}$ adjustment is needed to improve the estimation performance at high K_D levels.

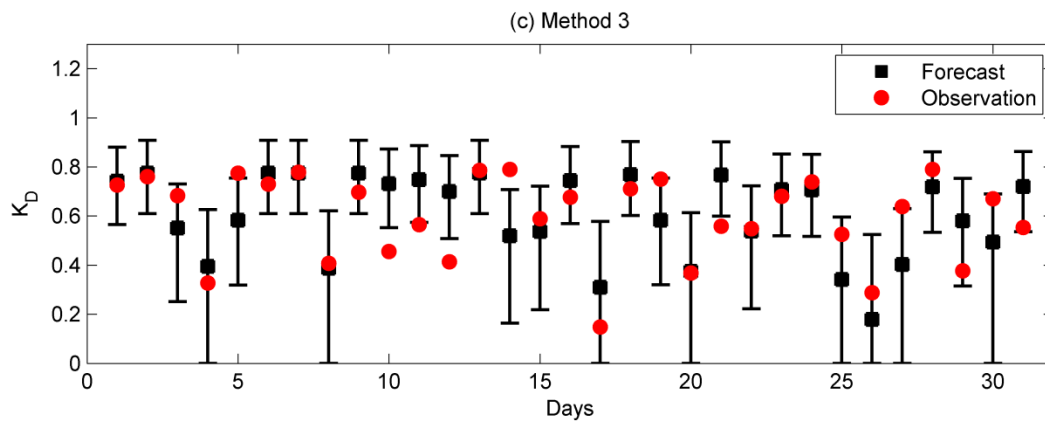
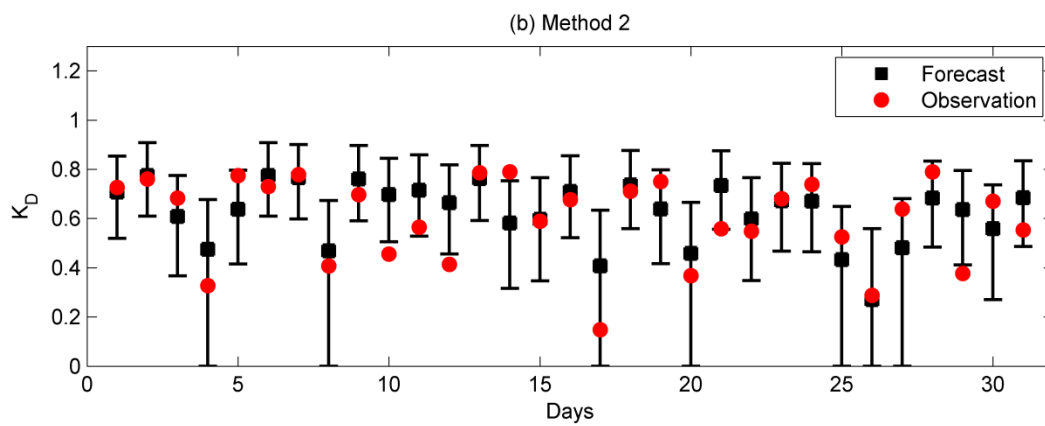
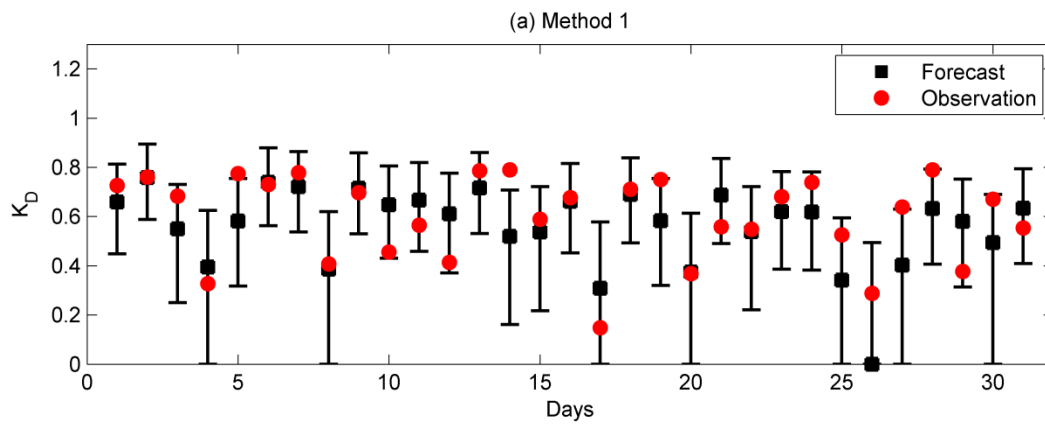
Results indicate that the adaptive $TKC_{D,bias}$ adjustment at different K_D levels (Methods 3, 4, and 5) is more effective than the single-value adjustment (Method 2). By using different adjustments for different K_D levels, Methods 3, 4, and 5 show improvements at the high K_D level without significant deterioration at the other levels. Considering all factors, Method 4 is better than Method 3, and Method 5 demonstrates the best performance especially in terms of KSI and OVER.

Fig. 4.13 displays the observed K_D , the $K_{D,for}$, and the calculated CIs of the $K_{D,for}$ for the five methods for the first 31 days of the tested year. If the observed K_D is within the CI of $K_{D,for}$, the estimation is considered to be satisfactory (a hit). Table 4.8 summarizes the percentage of the days that the observed K_D is within the CI interval for the entire tested year and in the days with

different K_D levels. The hit rate for the five methods ranges from 85.80% to 91.01% on the overall result.

Table 4.8: Percentage of days that the observed K_D is within the envelope of the proposed quantity estimation methods at different K_D levels for one year overall (2012).

Method	Overall (%)	High K_D (%)	Medium K_D (%)	Low K_D (%)
1	89.57	89.67	90.99	80.95
2	91.01	97.18	81.98	76.19
3	85.80	91.08	76.58	80.95
4	88.99	91.08	86.49	80.95
5	89.28	91.55	86.49	80.95



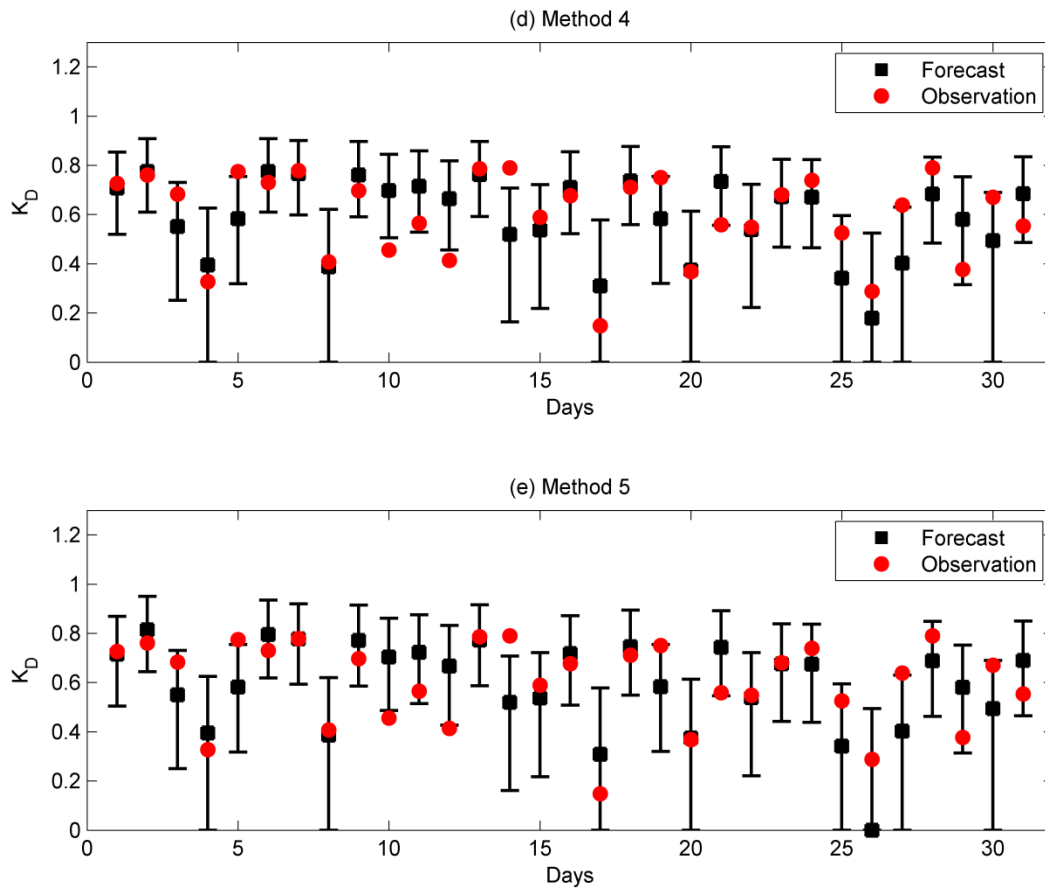


Fig. 4.13: Comparison of the estimated and observed K_D with the 95% CIs for the first 31 days of the year 2012 using (a) Method 1, (b) Method 2, (c) Method 3, (d) Method 4, and (e) Method 5, respectively.

4.7. Chapter contributions

This chapter provides several contributions. Firstly, this chapter quantifies the uncertainty for the estimation process. The NWS sky cover forecast shows approximately 20% uncertainty. This amount is identified as the main source of uncertainty for the overall estimation process. Secondly, the chapter demonstrates the need to adjust the NWS sky cover forecasts. Verification of the sky cover forecast reveals an overall overestimated bias. This bias is stronger at high irradiance levels. Thus, the estimation method needs to adaptively handle this bias at different irradiance levels. Thirdly, this chapter describes the conversion from cloud information to surface solar irradiance with a new equation. This conversion equation was developed based on the multi-year relationship between the sky imager-based cloud observation and the measured solar irradiance. The proposed conversion achieves accuracy with simplicity. Lastly, this chapter introduces five estimation methods for day-ahead solar irradiance using the NWS total sky cover forecast. These methods incorporate different schemes for dealing with the aforementioned bias of the cloud forecast. The observed data are regularly found within the 95% CIs of the estimated values.

CHAPTER 5. DAY-AHEAD QUALITY ESTIMATION OF SOLAR IRRADIANCE USING SKY COVER FORECAST

5.1. Chapter introduction

This chapter proposes a new methodology for estimating day-ahead solar irradiance quality using POP- K_D based on the day-ahead NWS sky cover forecast. The proposed method estimates day-ahead cloud fluctuation information from the NWS sky cover forecast, and the estimated cloud fluctuation information is subsequently used to estimate quality of solar irradiance at the ground level. Data from the SRRL BMS had been used for model development and result verification at different stages.

The rest of this chapter is organized as follows. Section 5.2 presents the details about observed and forecast sky cover data and measured solar irradiance data in the selected station. Section 5.3 describes that how POP- K_D represents the quality of daily solar irradiance. Section 5.4 discusses the relationship between quality of forecast sky cover and quality of observed sky cover. Section 5.5 describes the conversion from quality of sky cover to quality of solar irradiance. Section 5.6 proposes an estimation method. Section 5.7 shows the test results of the proposed method. Section 5.8 presents contributions of this chapter.

5.2. Data

As similar to the data acquisition in Chapter 4, ground-based observation of total sky cover for the SRRL BMS were retrieved from MIDC from 2009 through 2012 (MIDC, 2012). The historical total sky cover forecast data released from NWS were also retrieved from NDFD from 2009 through 2012 for forecast cloud information (NDFD, 2012).

For the daily POP- K_D calculation, I_{GHI} measured by the Kipp and Zonen CM22 pyranometer in 1-minute intervals over 12 years (2001–2012) at SRRL BMS were retrieved from MIDC (MIDC, 2012). Data for I_{GHI} were filtered from sunrise to sunset for each day, and then daily POP- K_D value was calculated based on the daily filtered I_{GHI} sequence.

5.3. Solar irradiance quality

5.3.1. Meaning of POP- K_D

As proposed in Chapter 3, POP- K_D is the POP observation based on a daily per-minute k dataset. POP observes differences between neighboring elements rounded by a threshold in the dataset and calculates the probability that these differences are equal to zero. Thus, POP- K_D is the probability over one day that the per-minute k will remain within the threshold from one time interval to the next. For example, if daylight hours are from 8 a.m. to 6 p.m. (10-hour duration), total per-minute time intervals come to 600 minutes. POP- K_D of 0.9 means that the per-minute k s are unchanged or changed within the threshold during 540 ($= 0.9 * 600$) of these time intervals.

Fig. 5.1 shows examples of how the per-minute k remains within the threshold during 30 minutes. For clear sky, the per-minute k is high and close to 1 depending on the atmospheric condition. Due to changes in the atmosphere such as cloud movement or cloud formation, the per-minute k fluctuates. POP- K_D indicates how often the per-minute k changes with a magnitude not exceeding the threshold. The per-minute k s in Fig. 5.1a show relatively less exceeding the threshold than the per-minute k s in Fig. 5.1b. Thus, POP- K_D in Fig. 5.1a is expected to be higher than that in Fig. 5.1b.

POP- K_D should be considered together with the associated threshold. For larger magnitude of the threshold, POP- K_D tends to be high (closer to 1.0). For a particular threshold, POP- K_D provides a statistical prediction for the frequency of fluctuation of solar irradiance. The threshold provides the magnitude of fluctuation for this frequency. POP- K_D is not very meaningful without the associated magnitude. A threshold of 0.1 was used here and Section 5.3.3 will show why the threshold of 0.1 is appropriate in this study.

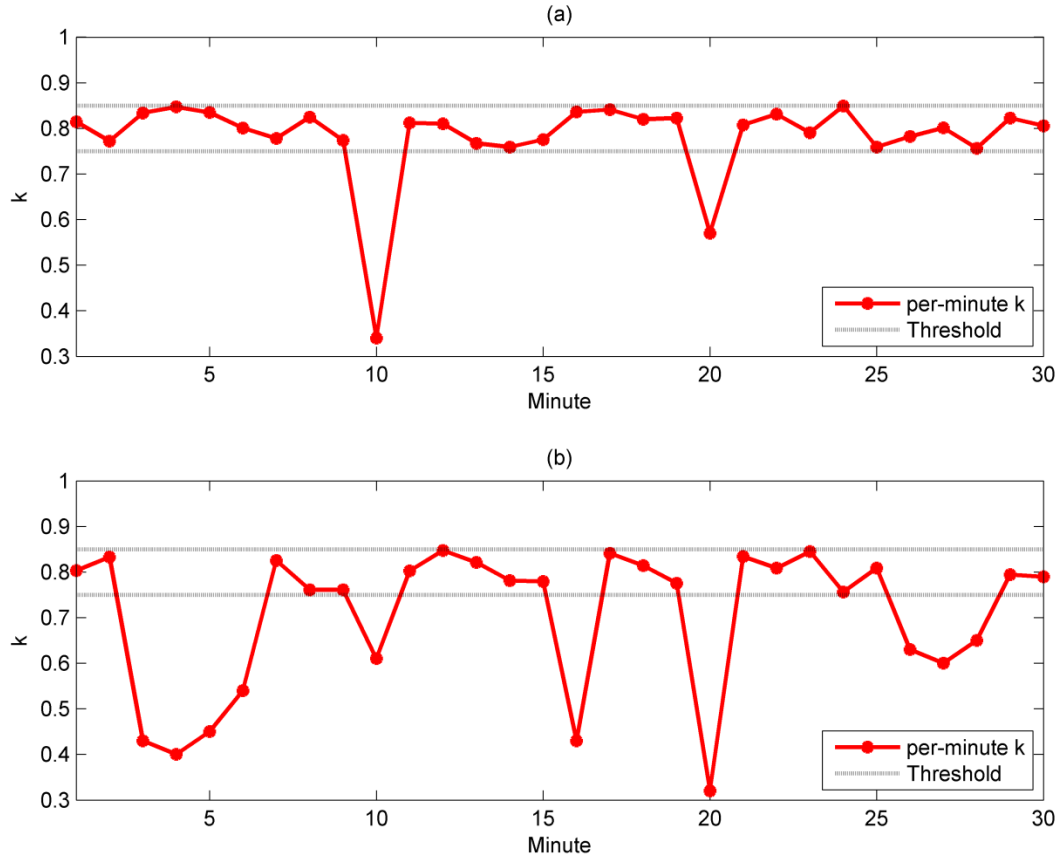
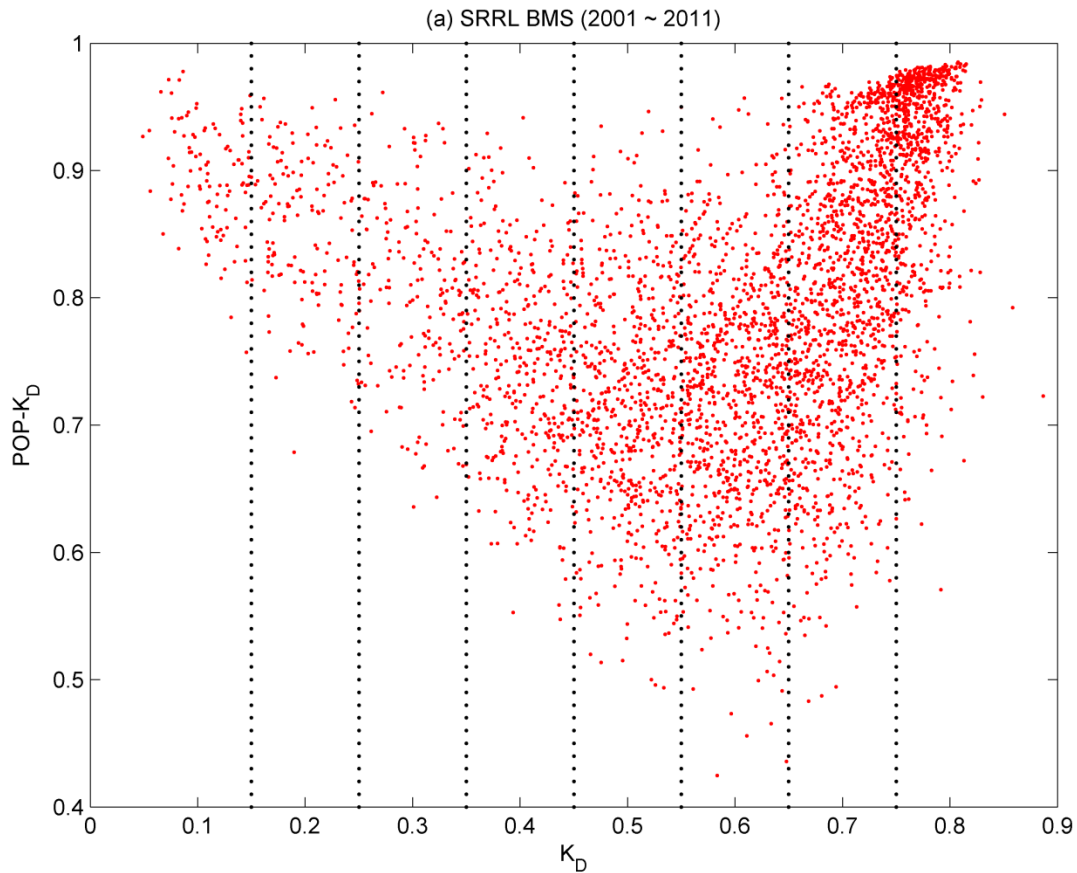


Fig. 5.1: Examples of how the per-minute k remains within the POP- K_D threshold: (a) relatively lower fluctuation and (b) relatively higher fluctuation.

5.3.2. Statistics of POP- K_D

The relationship between K_D and POP- K_D was observed for 11 years (2001 – 2011) at SRRL BMS. Fig. 5.2a shows cumulative data points of POP- K_D for 11 years at different K_D . As introduced in Chapter 3, the boundary of all data points in Fig. 5.2a can be enclosed by a specific polygon. This polygon shape indicates that the range of POP- K_D is the largest when K_D is around 0.6 and POP- K_D is getting smaller as K_D increases and decrease.

Fig. 5.2b shows boxplots of 8 different groups (Groups 1–8) of POP- K_D separated by different K_D ranges. Each boxplot is a display of statistical measures such as median and quartiles for a population. The comparison between the boxplots shows statistical variations between different populations. The upper and lower adjacent values were calculated by 1.5 interquartile range (IQR), which covers 99.3% of the data and excludes most extreme values. Group 1 is when $K_D < 0.15$. Groups 2–7 are increased in steps of 0.1 from 0.15 to 0.75. Group 8 is when $K_D \geq 0.75$. Table 5.1 summarizes the boxplot result of all 8 groups.



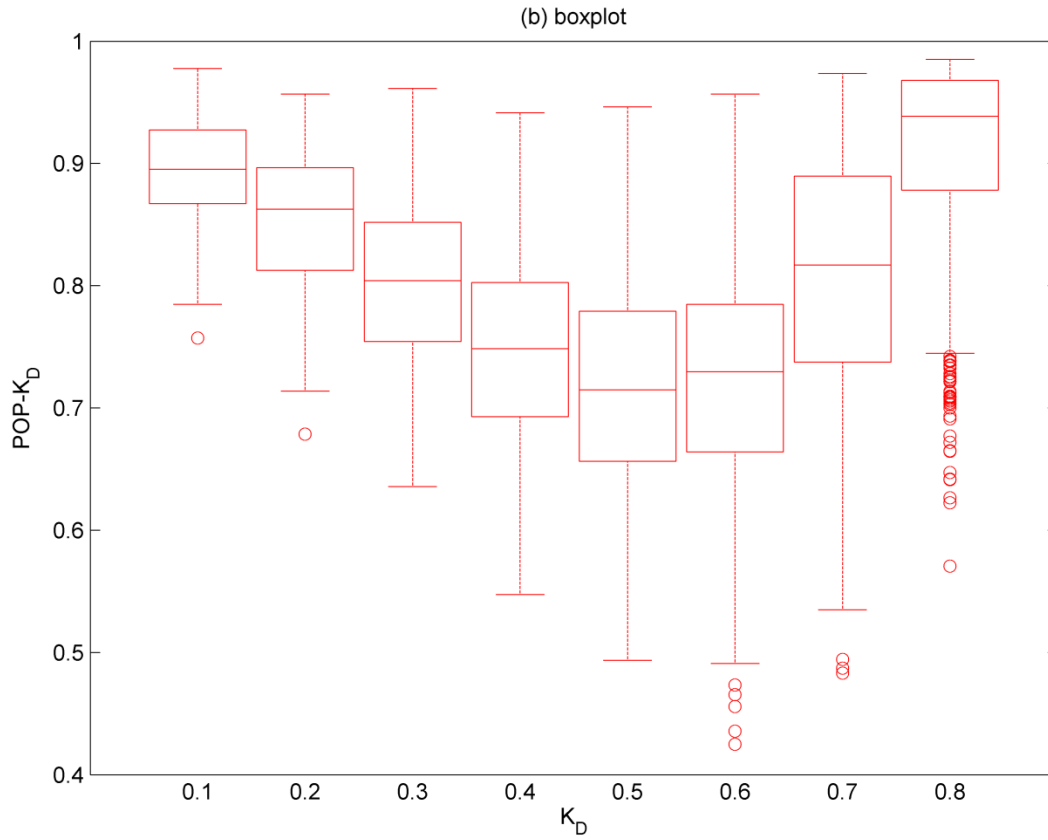


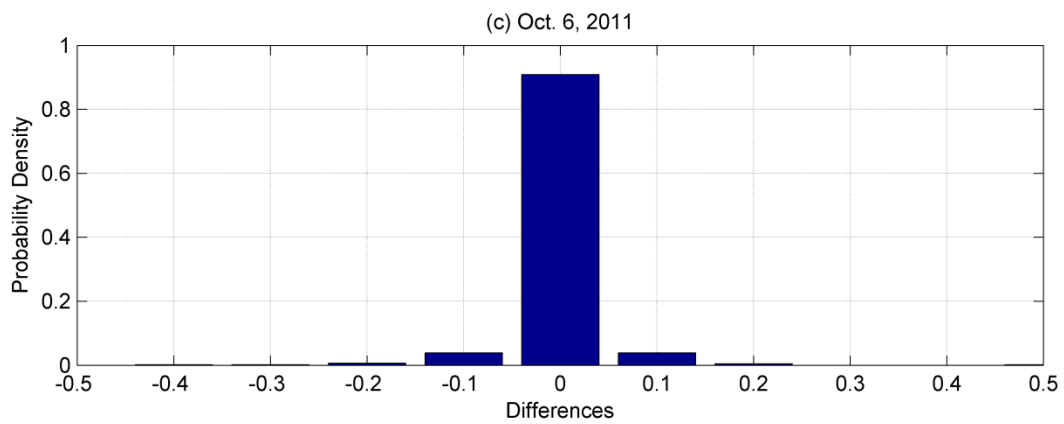
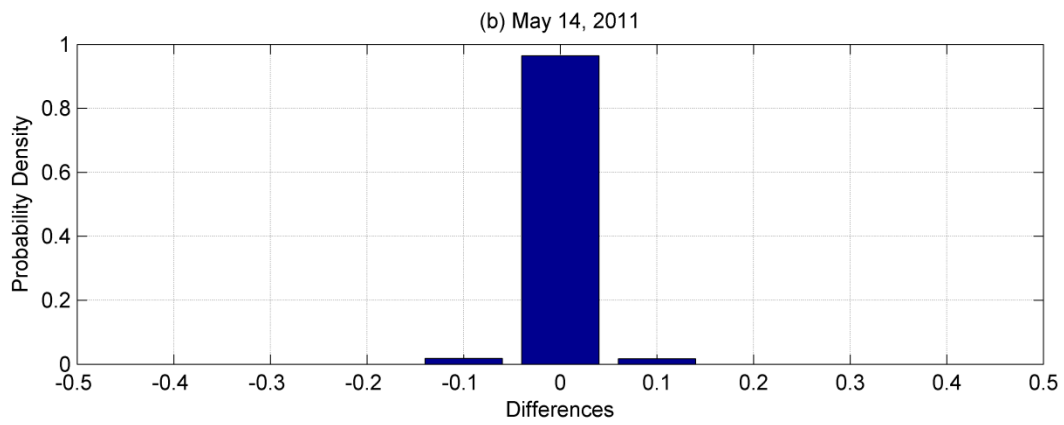
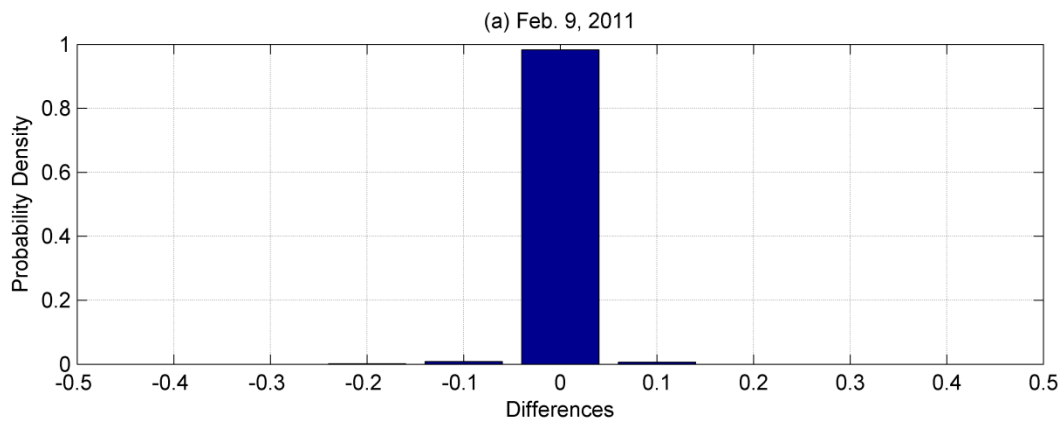
Fig. 5.2: Statistical relationship between K_D and $POP-K_D$ for 11 years (2001 – 2011) at SRRL

BMS; (a) cumulative data points and (b) boxplots grouped by different K_D .

Based on Table 5.1, the range between the upper and lower adjacent values is relatively wide at Groups 4 – 7 ($0.35 \leq K_D < 0.75$). Contrarily, Groups 1 ($K_D < 0.15$) and 8 ($K_D \geq 0.75$) show the narrowest (0.1931) and the second narrowest (0.2405) ranges, respectively. In addition, the medians of Groups 1 and 8 are bigger than them of Groups 5 and 6. This indicate that the fluctuation (quality) of daily solar irradiance is more certain (small variation) and relatively high (high median) at very high K_D level (Group 8) or very low K_D level (Group 1). However, the fluctuation is relatively lower and varying (wide range) at medium K_D levels (Groups 4–7).

Table 5.1: Boxplot result of POP- K_D grouped by different K_D ranges.

Group	1	2	3	4	5	6	7	8
K_D	$0 \leq K_D$	$0.15 \leq K_D$	$0.25 \leq K_D$	$0.35 \leq K_D$	$0.45 \leq K_D$	$0.55 \leq K_D$	$0.65 \leq K_D$	$0.75 \leq K_D$
Range	< 0.15	< 0.25	< 0.35	< 0.45	< 0.55	< 0.65	< 0.75	
Num. of POP- K_D	110 (2.51%)	151 (3.45%)	195 (4.45%)	406 (9.27%)	622 (14.20%)	868 (19.82%)	1289 (29.43%)	739 (16.87%)
Lower adjacent	0.7846	0.7137	0.6358	0.5475	0.4937	0.4911	0.5349	0.7446
Lower quartile	0.8672	0.8124	0.7543	0.6930	0.6563	0.6640	0.7376	0.8783
Median	0.8953	0.8626	0.8041	0.7484	0.7149	0.7297	0.8168	0.9387
Upper quartile	0.9275	0.8966	0.8517	0.8026	0.7791	0.7847	0.8896	0.9680
Upper adjacent	0.9777	0.9568	0.9613	0.9415	0.9463	0.9568	0.9735	0.9851



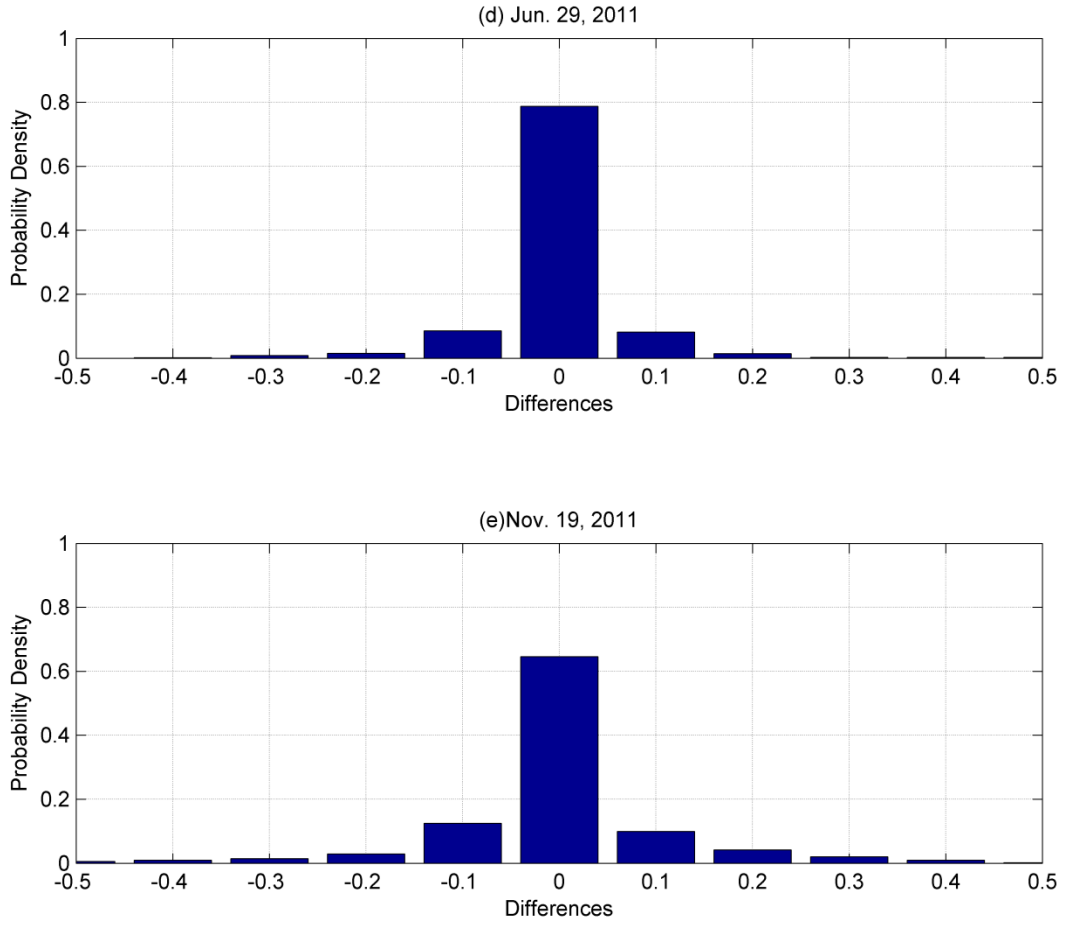


Fig. 5.3: Examples of daily probability density function (PDF) of $\Delta\tilde{k}$ at SRRL BMS in case of (a) high K_D , (b) low K_D , (c) medium K_D with high POP- K_D , (d) medium K_D with medium POP- K_D , and (e) medium K_D with low POP- K_D .

Table 5.2: K_D , POP- K_D , standard deviation, and kurtosis of the examples in Fig. 5.3.

Date	Figure	Group	K_D	POP- K_D	Standard deviation	Kurtosis
Feb. 9, 2011	Fig. 4a	8	0.8284	0.9830	0.0148	86.70
May 14, 2011	Fig. 4b	1	0.1204	0.9647	0.0188	28.31
Oct. 6, 2011	Fig. 4c	6	0.6452	0.9083	0.0442	44.91
Jun. 29, 2011	Fig. 4d	5	0.5377	0.7876	0.0852	21.86
Nov. 19, 2011	Fig. 4e	7	0.7053	0.6456	0.1143	8.502

Next, The daily probability density function (PDF) of $\Delta\tilde{k}$, the per-minute differences of k s rounded by the threshold, was observed for 11 years (2001–2011). The density at $\Delta\tilde{k} = 0$ indicates POP- K_D by the definition of POP- K_D . Fig. 5.3 shows five examples of the PDFs at different K_D levels. Table 5.2 compares K_D , POP- K_D , standard deviation, and kurtosis of the distributions in Fig. 5.3. A single value of the standard deviation represents the distributional uncertainty. The kurtosis indicates a peakedness of the distribution at mean and a fatness of the distribution at tails. Fig. 5.3a is the example in case of very high K_D ($K_D = 0.8284$). This distribution shows extremely high concentration at $\Delta\tilde{k} = 0$ and overall small variation. Thus, its standard deviation and kurtosis are relatively small and high, respectively. Fig. 5.3b is the example in case of very low K_D ($K_D = 0.1204$). The distributional shape is similar to that in Fig. 5.3a; the density at $\Delta\tilde{k} = 0$ is very high, and the variation is very small. The Kurtosis is relatively lower than Fig. 5.3a. Figs. 5.3c, 5.3d, and 5.3e show the examples in case of medium K_D level (Groups 4–7) with relatively high POP- K_D , medium POP- K_D , and low POP- K_D ,

respectively. These three PDFs are more widely distributed than Figs 5.3a and 5.3b. The distribution in Fig. 5.3c shows relatively higher concentration at $\Delta\tilde{k} = 0$ and lower variation compared to the distributions in Figs. 5.3d and 5.3e. Contrarily, the distribution in Fig. 5.3e shows relatively lower concentration at $\Delta\tilde{k} = 0$ and higher variation.

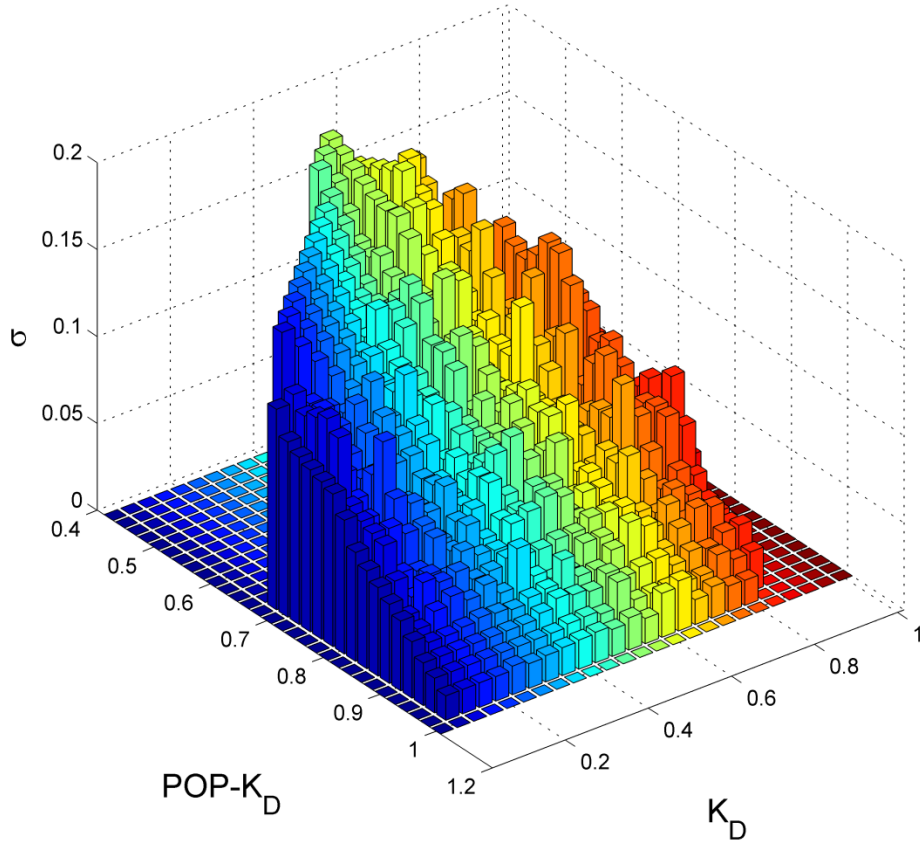


Fig. 5.4: 3-D synoptic view of K_D , $POP-K_D$, and standard deviation (σ) at SRRL BMS for 11 years (2001 – 2011).

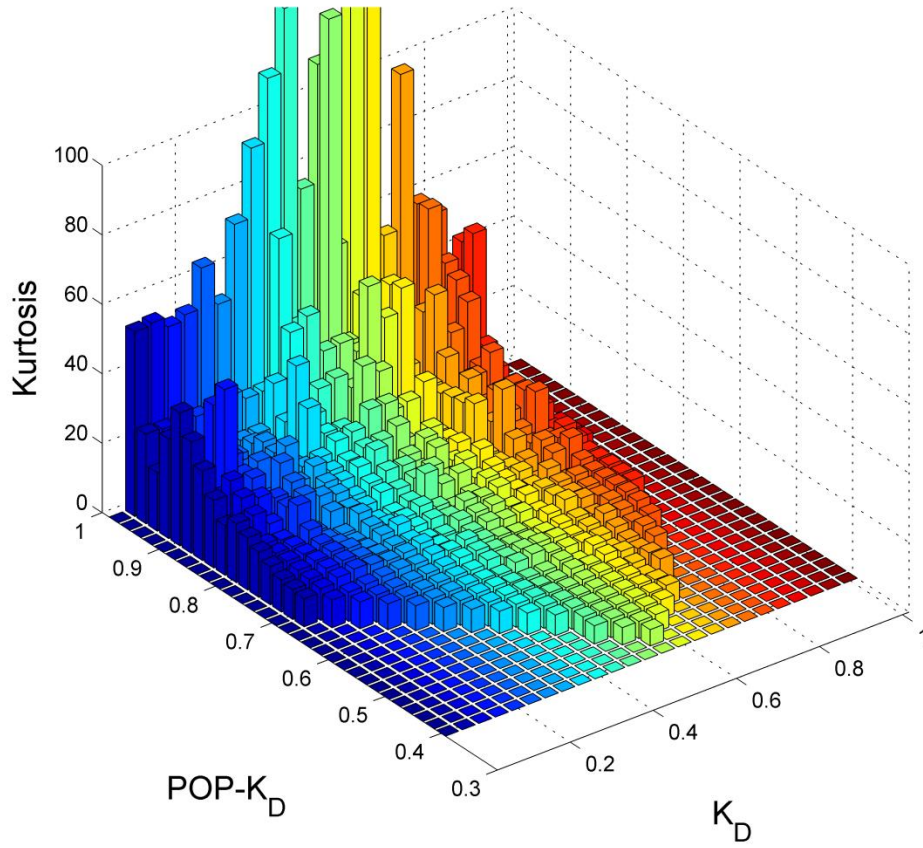


Fig. 5.5: 3-D synoptic view of K_D , $POP-K_D$, and kurtosis at SRRL BMS for 11 years (2001 – 2011).

Fig. 5.4 shows a three-dimensional synoptic view of K_D , $POP-K_D$, and standard deviation for 11 years. The two-dimensional shape between K_D and $POP-K_D$ shows the specific polygon boundary shown in Fig. 5.2a. The standard deviation values are placed on this polygon as bars. Fig. 5.4 shows that the standard deviation generally decreases as $POP-K_D$ increases for all K_D . A single value of the standard deviation represents the distributional uncertainty. Thus, this shows that a higher $POP-K_D$ promises a higher certainty for the daily distribution.

Fig. 5.5 shows the three-dimensional synoptic view of K_D , POP- K_D , and kurtosis for 11 years. Fig. 5.5 shows that all distributions are positive excess kurtosis (leptokurtic) since all kurtoses are larger than 3. In addition, the kurtosis generally increases as POP- K_D increases for all K_D . Distributions with the excess high kurtosis depict higher concentration (peakedness) at mean. Thus, the result in Fig. 5.5 demonstrates that a higher POP- K_D has more chances that the per-minute fluctuation is equal to zero. However, as the leptokurtic distribution includes fatter tails compared to a normal distribution, a higher POP- K_D may have more chances that the fluctuation is more volatile than the normal distribution.

Based on the results in Figs. 5.4 and 5.5, one quantity of POP- K_D describes both these characteristics (uncertainty and kurtosis) of the distribution of the per-minute differences in daily solar irradiance. Thus, this indicates that POP- K_D is useful for the quality measure of daily solar irradiance

5.3.3. Ramp rate

Fig. 5.6 shows cumulative distribution functions (CDFs) of per-minute changes (ramp rates) of I_{GHI} for 11 years (2001 – 2011) at SRRL BMS: (a) entire CDFs and (b) graphs zoomed between 0 and 150W/m² for x-axis and between 0.8 and 1 for y-axis. The distributions were classified into different day types. The day type can be classified via K_D into high ($K_D \geq 0.6$), medium ($0.3 \leq K_D < 0.6$), and low irradiance levels ($K_D < 0.3$) as introduced in Chapter 3. In Fig. 5.6, the ramp rates on low K_D days show a relatively higher cumulative frequency at the lower magnitude due to the relatively smaller fluctuation of I_{GHI} . Comparatively, the ramp rates on medium K_D days show a relatively lower cumulative frequency due to the larger variability of I_{GHI} .

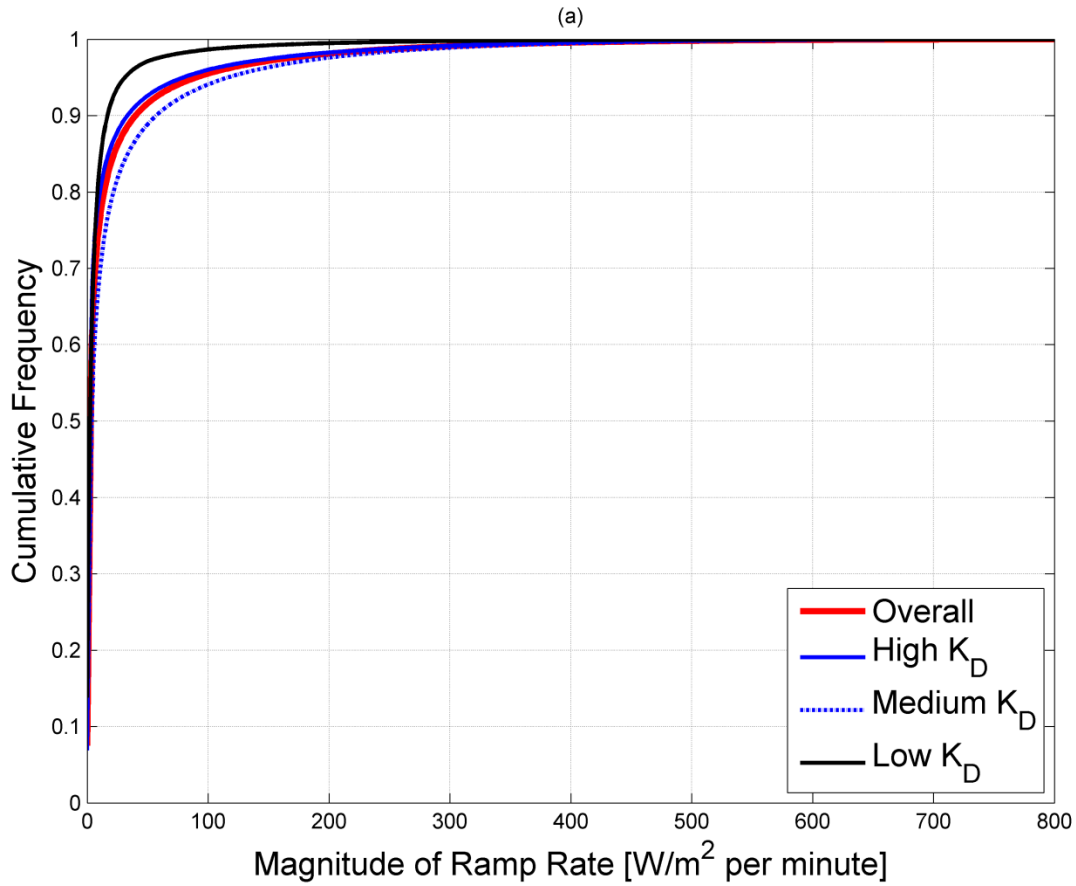
If the maximum clear sky irradiance at ground level is $1,000 \text{ W/m}^2$, the ground-based sky clearness index (k_{GND}) can be defined by modifying Eq. (3.1) as:

$$k_{GND} = \frac{I_{GHI}}{1,000 \cdot E_0 \cdot \cos \theta_Z} . \quad (5.1)$$

k_{GND} provides a ratio of I_{GHI} compared to the maximally available irradiance at ground level.

Based on Eqs. (3.1) and (5.1), the relationship between k and k_{GND} can be derived as:

$$k_{GND} = 1.361 \cdot k . \quad (5.2)$$



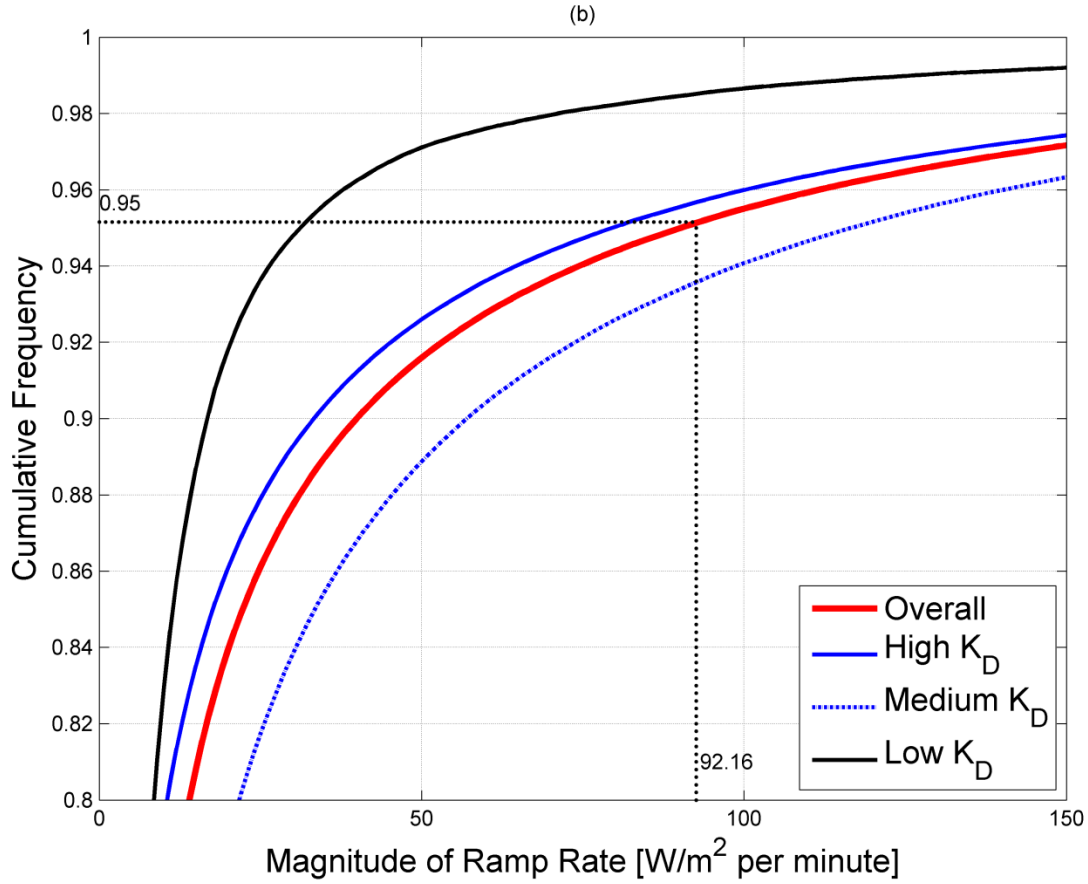


Fig. 5.6: CDFs of ramp rates of I_{GHI} for overall 11-year (2001 – 2011) observation at SRRL BMS and for days with high K_D , medium K_D , and low K_D levels: (a) entire CDFs and (b) graphs zoomed between 0 and 150W/m² for x-axis and between 0.8 and 1 for y-axis.

The threshold of 0.1 was used for calculating POP- K_D . This corresponds to $\pm 5\%$ per minute in proportion to I_{ETI} based on Eq. (3.1). Since the maximum value of I_{ETI} is I_0 (1,361W/m²), the maximum range of $\pm 5\%$ in k is $\pm 68.05\text{W/m}^2$ ($=5\% \cdot I_0$). This represents the ramp rate, and this corresponds to $\pm 92.16\text{W/m}^2$ ($=1.361 \cdot \pm 68.05$) in k_{GND} based on Eq. (5.2). As shown in Fig. 5.6b, 92.16W/m² includes approximately 95% of the ramp rates for the overall 11-

year period. This indicates that the threshold of 0.1 can cover 95% of the per-minute changes of I_{GHI} at ground level. This result shows why the threshold of 0.1 is suitable for the POP- K_D calculation at this location.

Since POP- K_D indicates the per-minute changes of solar irradiance within the threshold, POP- K_D can be related to the ramp rate control of generators in power systems. Using the assumption that the maximum clear sky irradiance is $1,000\text{W/m}^2$ as the rating, the ground-based maximum ramp rate by the threshold, 92.16W/m^2 , is about 9.22%. As the PV output is directly proportional to solar irradiance, 9.22% fluctuation of solar irradiance causes the same fluctuation of the PV output. If a PV penetration level is 20%, 9.22% change of the PV output affects 1.84% ($= 20\% * 9.22\%$) change for an entire system. Common generators in the system may compensate this 1.84% ramp rate because their ramp rate capability is about 2 – 5% per minute (Kirby and Milligan, 2005). Thus, POP- K_D could mean the probability that the solar irradiance variability is within the ramp rates of the common generators in power systems at a certain PV penetration rate.

5.4. Relationship between quality of observed sky cover and quality of forecast sky cover

5.4.1. Relationship between $POP-TKC_{D,NWS}$ and $POP-TKC_D$

The POP observation of total sky cover can be used for estimating $POP-K_D$ at ground level. The total sky cover can be determined from sky imager-based observation and the NWS sky cover forecast. Thus, this research distinguishes between the observed total sky cover, defined as TKC , and the forecast total sky cover, defined as TKC_{NWS} .

$POP-TKC_D$ is defined as the POP observation of the daily dataset of TKC observed by TSI. $POP-TKC_{D,NWS}$ is defined as the POP observation of an hourly sequence of TKC_{NWS} . The details about retrieval and processing of the hourly sequence of TKC_{NWS} were presented in Chapter 4.

In describing identical time series for sky cover, TKC_{NWS} is predicted at 1-hour intervals while TKC is observed at 1-minute intervals. The different temporal intervals for the POP calculation affect different ranges of the calculated POP values. $POP-TKC_{D,NWS}$ and $POP-TKC_D$ are expected to be similar on days with high persistence because most ΔTKC_{NWS} and ΔTKC are zero. However, $POP-TKC_D$ is expected to be higher than $POP-TKC_{D,NWS}$ on days with partial or full fluctuation because the minute-to-minute difference (ΔTKC) tends to be more consistent than the hourly difference (ΔTKC_{NWS}).

Fig. 5.7a shows CDFs of $POP-TKC_{D,NWS}$ and $POP-TKC_D$ based on a 3-year dataset (2009 – 2011). $POP-TKC_{D,NWS}$ is distributed from 0.1250 to 1 while $POP-TKC_D$ is distributed from 0.6737 to 1. The maximum value for each of the two parameters is identical, but the minimum value is significantly different.

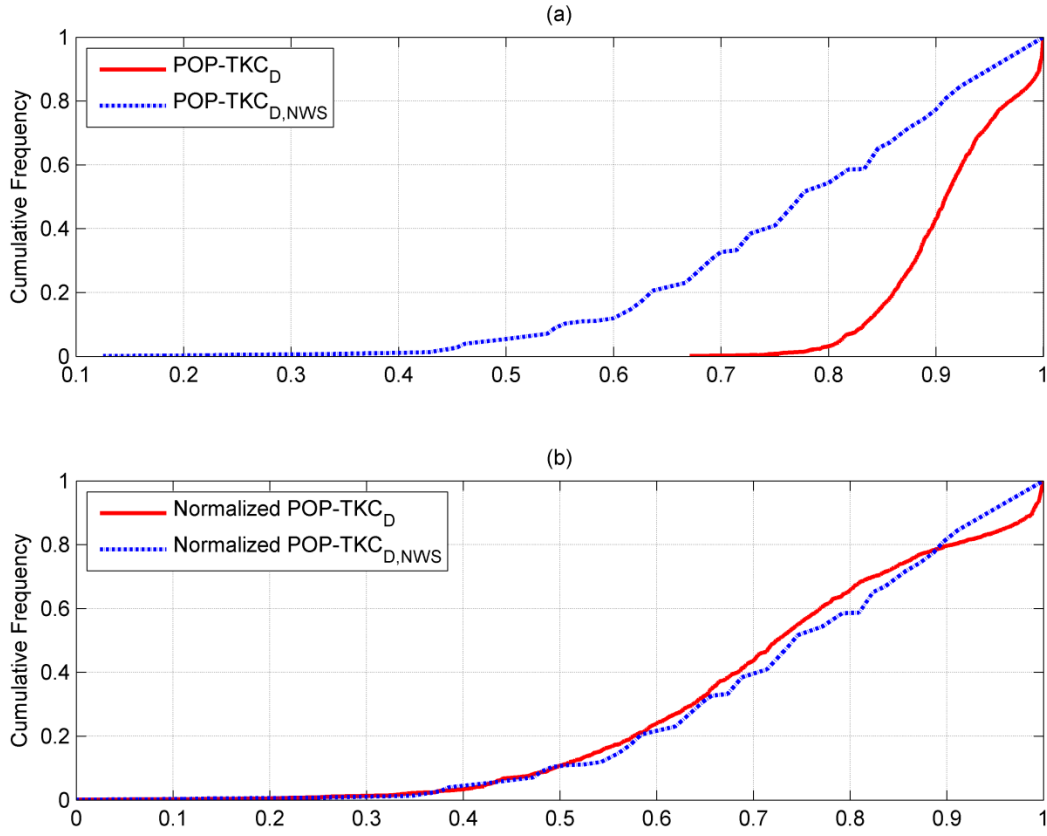


Fig. 5.7: Comparison of CDFs of POP-TKC_D and POP-TKC_{D,NWS} for 3-year (2009 – 2011): (a) original values and (b) normalized values.

The different ranges between POP-TKC_{D,NWS} and POP-TKC_D can be located on an identical data range (between 0 and 1) using normalization by range. The normalization by range scales all data points in the range between 0 and 1. The newly normalized value, X_{nor} , is calculated by:

$$X_{nor} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}, \quad (5.3)$$

where X is the original data point, and X_{\min} and X_{\max} are the minimum and maximum values for all data points, respectively. Fig. 5.7b shows the CDFs of normalized POP-TKCD,NWS and POP-TKCD. Compared to Fig. 5.7a, the two normalized parameters in Fig. 5.7b are similarly distributed in $[0, 1]$.

With the assumption that the normalized POP-TKCD,NWS and POP-TKCD are equal, the relationship between POP-TKCD,NWS and POP-TKCD can be derived by using Eq. (5.3) as:

$$\text{POP-TKCD} = 0.3729 \text{ POP-TKCD,NWS} + 0.6271. \quad (5.4)$$

Based on Eq. (5.4), daily cloud fluctuation information (POP-TKCD) can be predicted using the NWS sky cover forecast data (POP-TKCD,NWS).

5.4.2. Verification

The accuracy of Eq. (5.4) was verified by computing $\Delta\text{POP-TKCD}$, defined as the difference between estimated POP-TKCD (POP-TKCD,for) from POP-TKCD,NWS and the corresponding POP-TKCD obtained from observation. Data for the year 2012 at SRRL BMS were obtained for this verification (data for 345 days; data for 21 days were missing). Table 5.3 summarizes the statistical errors, RMSE and MBE, of $\Delta\text{POP-TKCD}$ at different irradiance levels. The result in Table 5.3 shows that $\Delta\text{POP-TKCD}$ is generally small for the overall 1-year and for all K_D levels. This demonstrates the effectiveness of the normalization approach using Eq. (5.4).

Table 5.3: RMSE and MBE of $\Delta\text{POP-TKC}_D$ for 1 year (2012).

	Overall	High K_D	Medium K_D	Low K_D
RMSE	0.0662	0.0634	0.0721	0.0604
RMSE (%)	7.18%	6.78%	8.07%	6.48%
MBE	-0.0034	0.0119	-0.0163	0.0216
MBE (%)	-0.37%	1.28%	-1.83%	2.32%

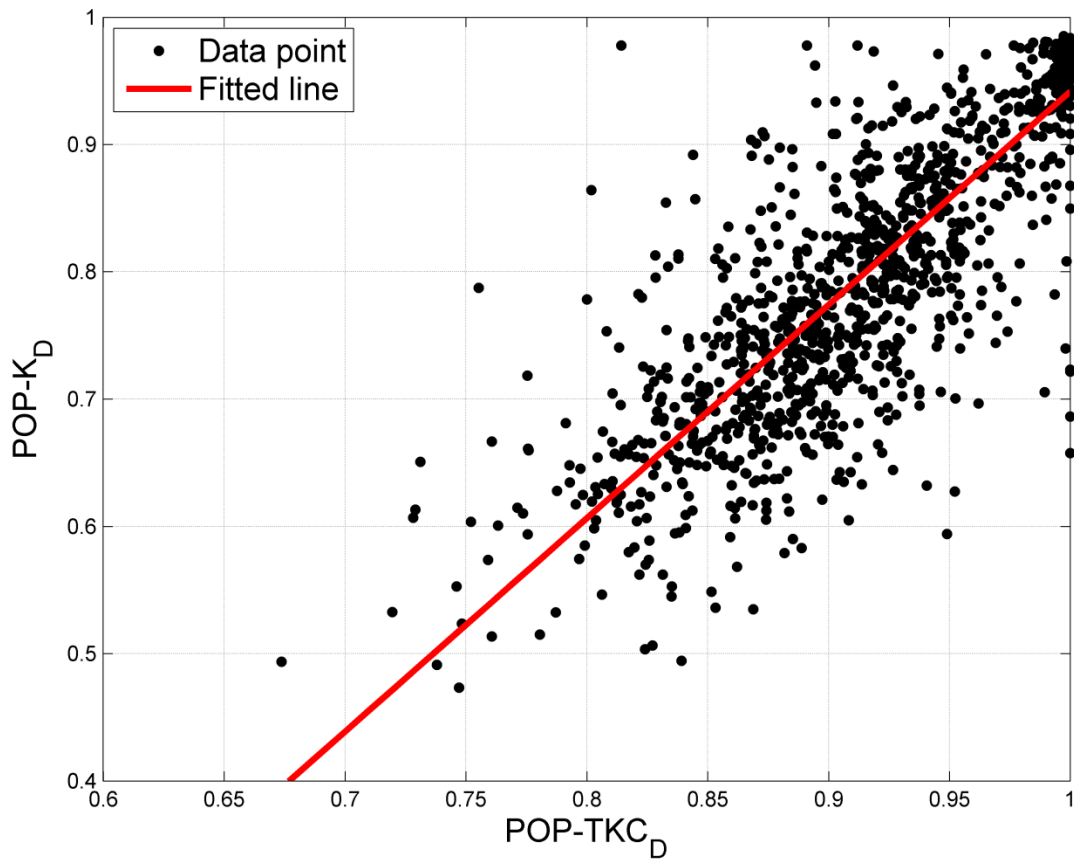


Fig. 5.8: Relationship between POP-TKC_D and POP-K_D at SRRL BMS for 3 years (2009 – 2011).

5.5. Relationship between quality of observed sky cover and quality of solar irradiance

5.5.1. Relationship between POP-TKC_D and POP-K_D

The relationship between POP-TKC_D and POP-K_D was established for estimating the daily fluctuation of solar irradiance using the daily fluctuation of total sky cover. Fig. 5.8 shows this relationship using data over a 3-year period (2009–2011).

The relationship in Fig. 5.8 can be represented by a linear equation using least square fitting with bisquare robustness as follows:

$$\begin{aligned} POP - K_D &= a_2 POP - TKC_D + b_2 \\ a_2 &= 1.677 [1.615, \quad 1.740] \\ b_2 &= -0.7353 [-0.7925, \quad -0.6780] \end{aligned} \tag{5.5}$$

where the estimated coefficients a_1 and b_1 are specified with their respective 95% CIs.

5.5.2. Verification

The accuracy of the POP-TKC_D-vs-POP-K_D equation, Eq. (5.5), was verified by the observation of Δ POP-K_D, defined as the comparison of observed POP-K_D and POP-K_D obtained from observed POP-TKC_D by using Eq. (5.5). The data for the year 2012 were applied for this purpose. Table 5.4 shows RMSE and MBE of Δ POP-K_D at different K_D levels. The statistical errors in Table 5.4 are generally minor for the overall 1-year and for all K_D levels. This result

indicates the usefulness of Eq. (5.5). Compared to the statistical results of $\Delta\text{POP-TKC}_D$ shown in Table 5.3, $\Delta\text{POP-K}_D$ shows slightly higher errors in general.

Table 5.4: RMSE and MBE of $\Delta\text{POP-K}_D$ for 1 year (2012).

	Overall	High K_D	Medium K_D	Low K_D
RMSE	0.0774	0.0735	0.0846	0.0755
RMSE (%)	9.58%	8.85%	11.32%	8.68%
MBE	0.0077	-0.0078	-0.0165	0.0400
MBE (%)	0.96%	-0.95%	-2.22%	4.61%

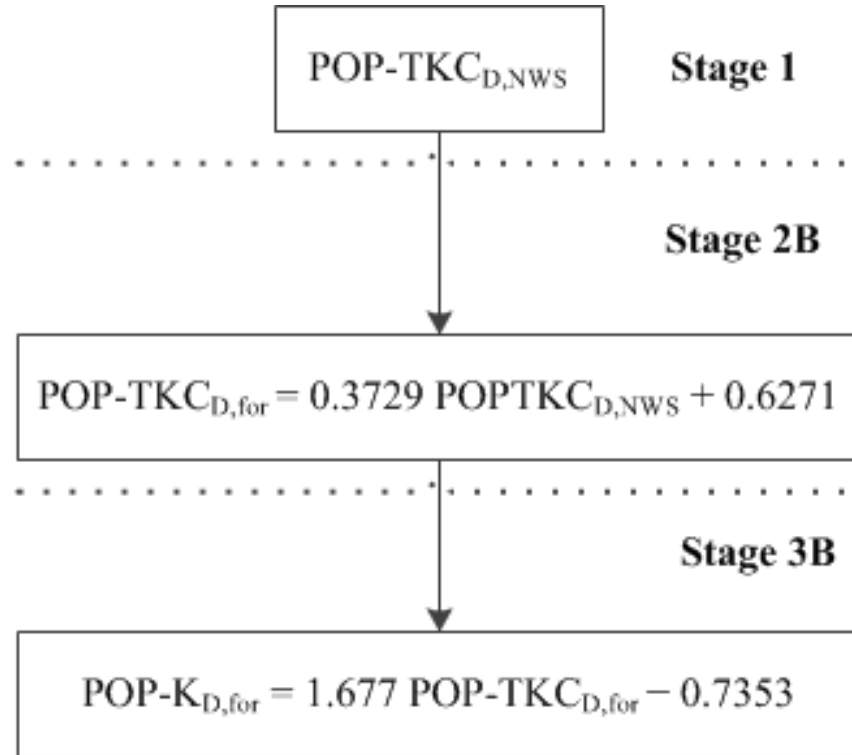


Fig. 5.9: Algorithm for the proposed quality estimation method.

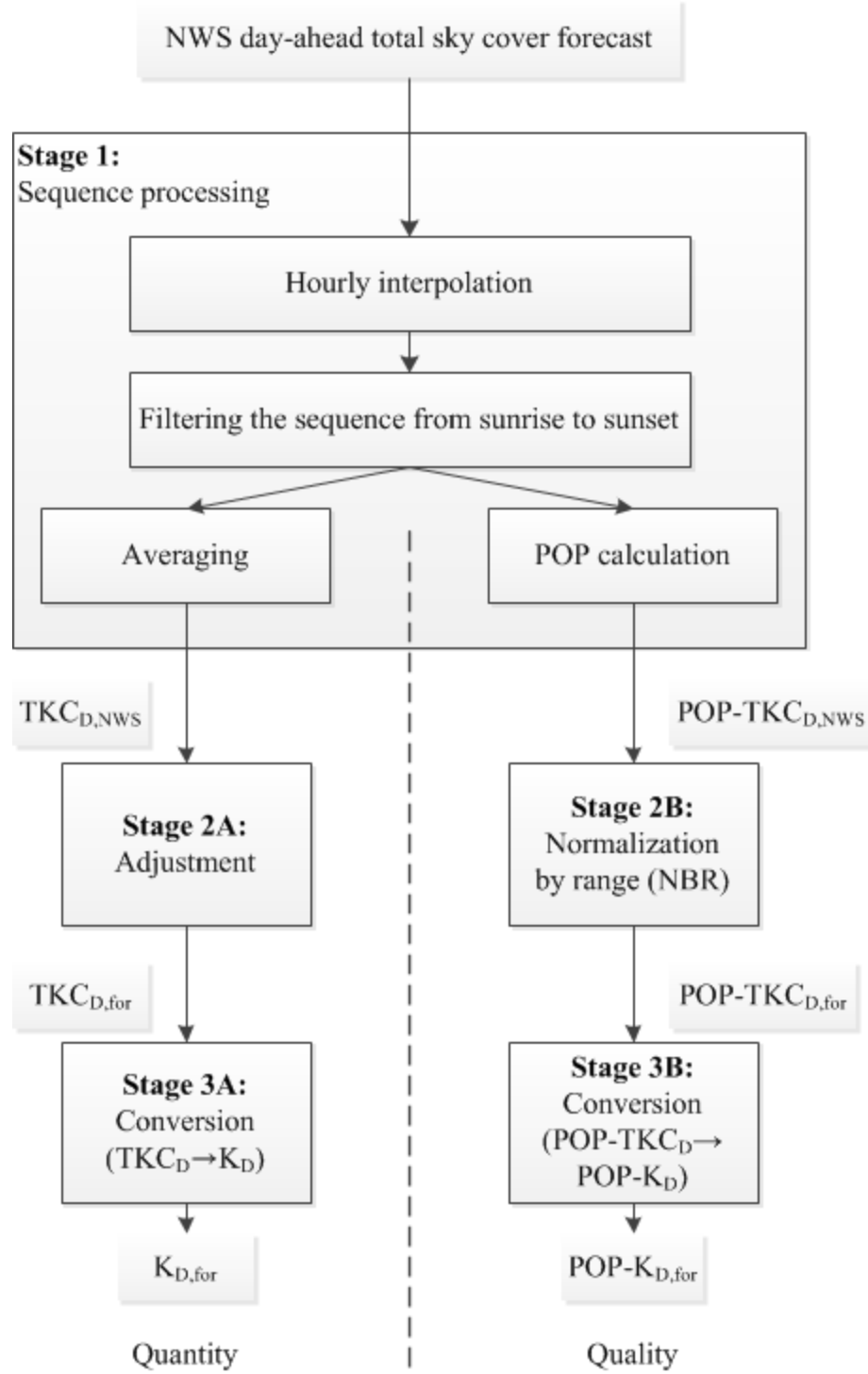


Fig. 5.10: Overall sequence of the proposed estimation methods for quantity and quality of daily solar irradiance.

5.6. Proposed estimation method for day-ahead POP-K_D

This chapter proposes a day-ahead POP-K_D estimation methods that are variations of the 3-stage sequence.

- Stage 1: Processing the NWS day-ahead total sky cover forecast and converts it into POP-TKC_{D,NWS} data based on the POP calculation.
- Stage 2B: Converting the POP-TKC_{D,NWS} data into POP-TKC_{D,for} via Eq. (5.4).
- Stage 3B: Estimating POP-K_{D,for} from POP-TKC_{D,for} based on Eq. (5.5).

Fig. 5.9 shows the algorithm for the proposed quality estimation method. Accordingly, Fig. 5.10 shows a block diagram of combination of the proposed quantity and quality methods.

The 95% CI of POP-K_{D,for} for this method is estimated as follows. The uncertainty of model-based day-ahead cloud forecasts can be estimated as about 15% (Shah et al., 2010). Thus, the uncertainty at the output of Stage 1 (uncertainty of POP-TKC_{D,NWS}), defined as σ_{1-B} , is assumed as 0.15 here.

For analyzing uncertainty in Stage 2B, the standard deviation of POP-TKC_{D,for} (σ_{2B}) can be calculated by using Eq. (5.4). σ_{2B} can be determined by the following equation:

$$\sigma_{2B} = \sqrt{(c_{1B} \sigma_{1B})^2} \quad (5.6)$$

where c_{1B} is a sensitivity coefficient of POP-TKC_{D,NWS} (Ellison and Williams, 2012). The sensitivity coefficient is determined by taking the partial difference of Eq. (5.4) as following:

$$c_{1B} = \frac{\partial(POP - TKC_{D,for})}{\partial(POP - TKC_{D,NWS})} = 0.3729. \quad (5.7)$$

Consequently, σ_{2B} is calculated as 0.05594.

Using Eq. (5.5), the standard deviation of POP- $K_{D,for}$ (σ_{3B}) is determined as following:

$$\sigma_{3B} = \sqrt{(c_{a2}\sigma_{a2})^2 + (c_{b2}\sigma_{b2})^2 + (c_{2B}\sigma_{2B})^2} \quad (5.8)$$

where c_{a2} , c_{b2} , and c_{2B} are sensitivity coefficients of a_2 , b_2 , and POP- $TKC_{D,for}$, respectively, and σ_{a2} and σ_{b2} are standard deviations of a_2 and b_2 , respectively. The c_{a2} , c_{b2} , and c_{2B} are calculated by:

$$c_{a2} = \partial(POP - K_{D,for}) / \partial a_2 = POP - TKC_{D,for}, \quad (5.9)$$

$$c_{b2} = \partial(POP - K_{D,for}) / \partial b_2 = 1, \quad (5.10)$$

$$c_{2B} = \partial(POP - K_{D,for}) / \partial (POP - TKC_{D,for}) = a_2 = 1.677. \quad (5.11)$$

Based on the CIs of a_2 and b_2 , σ_{a2} and σ_{b2} are achieved as:

$$\sigma_{a2} = |1.740 - 1.615| / (2 \cdot 1.96) = 0.03189, \quad (5.12)$$

$$\sigma_{b2} = |-0.6780 - (-0.7925)| / (2 \cdot 1.96) = 0.02921. \quad (5.13)$$

The variable c_{a2} depends on POP- $TKC_{D,for}$, but σ_{a2}^2 is small. Therefore, σ_{3B} has an approximately constant value of 0.1016 regardless of POP- $TKC_{D,for}$. Finally, the upper and lower boundaries of the 95% CI of POP- $K_{D,for}$ are obtained as following equations:

$$Upper(POP - K_{D,for}) = 1.677 (POP - TKC_{D,for}) - 0.5362, \quad (5.14)$$

$$Lower(POP - K_{D,for}) = 1.677 (POP - TKC_{D,for}) - 0.9344. \quad (5.15)$$

Fig. 5.11 describes these boundaries along with the conversion curve of Eq. (5.5). Since σ_{3B} is constant for all $TKC_{D,for}$ values, the CIs using Eqs. (5.14) and (5.15) show the linear line.

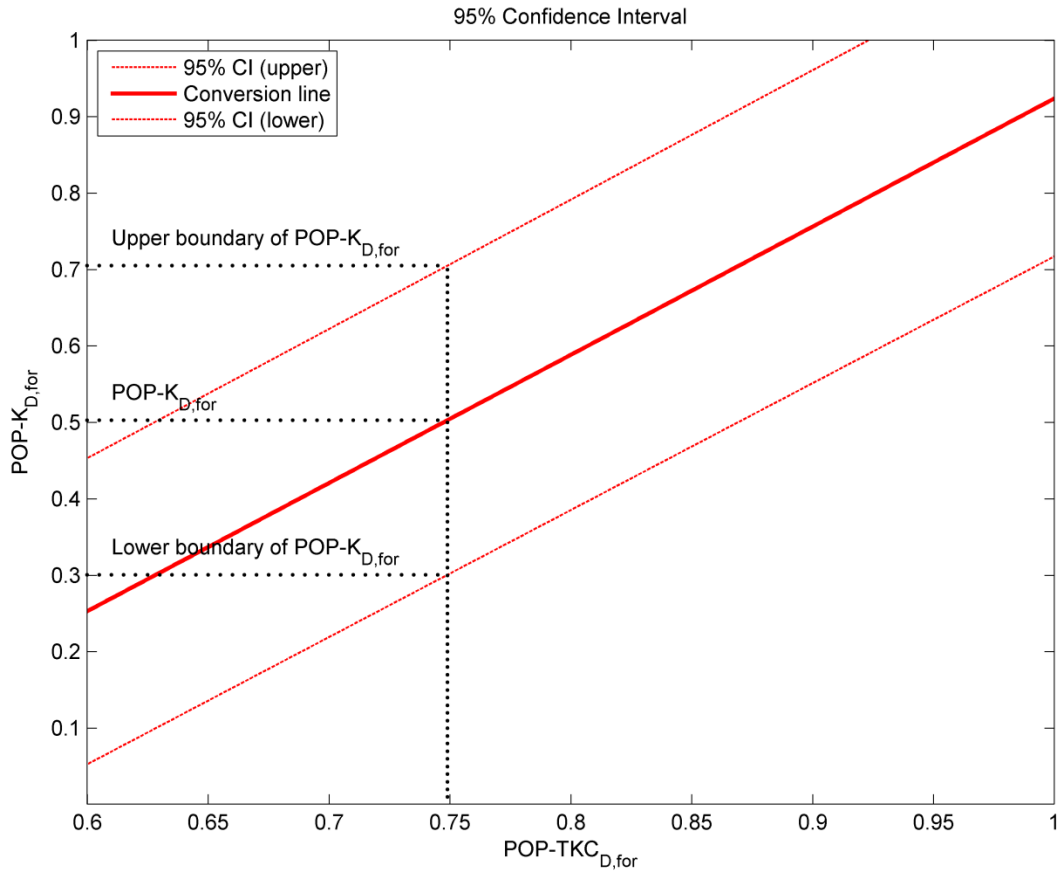


Fig. 5.11: Conversion lines and the boundaries for the 95% confidence interval (CI) for the proposed quality estimation method.

5.7. Results

The proposed quality estimation method was applied in order to predict POP- $K_{D,for}$ by using the POP-TKC $_{D,NWS}$ data for the year 2012 at SRRL BMS. The estimated values were compared to the corresponding values of the observed POP- K_D in 2012 for verification.

Table 5.5 summarizes the RMSE and MBE for the proposed method on different day types. The RMSE shows an approximately constant pattern for the year overall and at all K_D levels (15.35 – 17.15%). The highest error (17.15%) was reported at the medium K_D level due to the relatively high fluctuation in medium K_D as compared to the other two levels. The MBE for the year overall is significantly small (0.24%). The three different K_D levels show small errors (-5.90 – 8.79%). A negative bias (-5.90%) is reported at the medium K_D level. Thus, the proposed method tends to slightly underestimate on days when a higher fluctuation is expected.

Fig. 5.12a depicts the CDF of the estimation result from using the proposed quality method as well as the CDF of the observation data. Fig. 5.12b shows the differences between the CDF of the proposed method and the CDF of observed data over all ranges of POP- $K_{D,for}$. As shown in Fig. 5.12b, the proposed method mostly stays under the threshold but the difference are above the threshold at around POP- $K_{D,for} = 0.7$ and 0.9. The parameters of KSI and OVER were tested and the test result shows 54.83% for KSI and 3.49% for OVER.

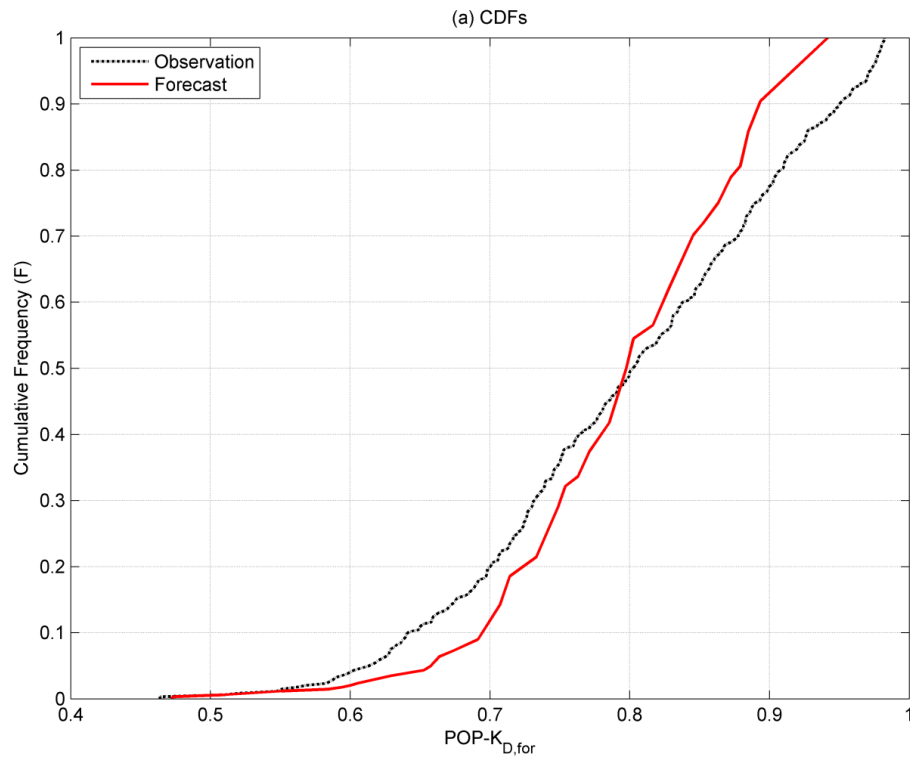
Fig. 5.13 displays the observed value and the 95% CIs of the estimated value for the proposed quality estimation method for the first 31 days of the tested year. If the observed POP- K_D value is within the CI of the estimated value, the estimation is considered to be satisfactory (a hit). Table 5.6 summarizes the percentage of days that the observed POP- K_D is within the CI interval for the entire tested year and on the days with different K_D levels. The hit rate for the proposed POP- K_D estimation method is shown to be more than 80% for the overall tested year.

Table 5.5: Comparison of RMSE and MBE of the proposed quality method for one year (2012)

	Overall	High K_D	Medium K_D	Low K_D
RMSE	0.1308	0.1319	0.1283	0.1335
RMSE (%)	16.19%	15.86%	17.15%	15.35%
MBE	0.0020	0.0122	-0.0438	0.0763
MBE (%)	0.24%	1.48%	-5.90%	8.79%

Table 5.6: Percentage of days that the observed POP- K_D is within the CIs of the proposed quality method (hit rate) for one year (2012).

	Overall	High K_D	Medium K_D	Low K_D
Hit rate (%)	81.45	81.22	81.98	80.95



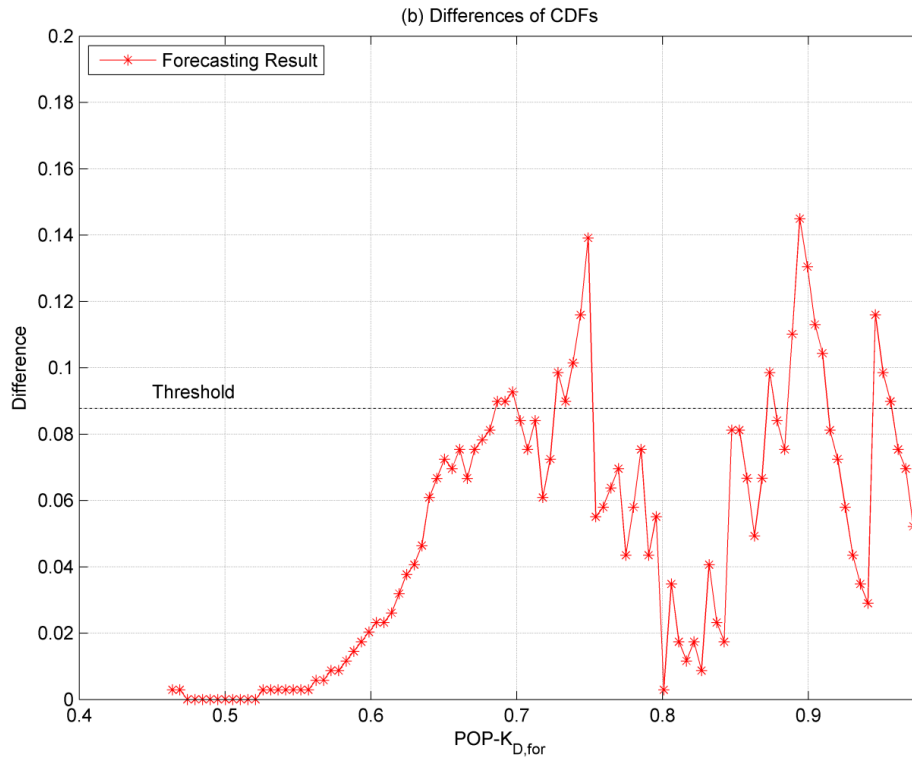


Fig. 5.12: K–S statistics for the proposed quality estimation method: (a) CDFs and (b) differences of the CDF generated by the proposed method and the CDF of the observed data.

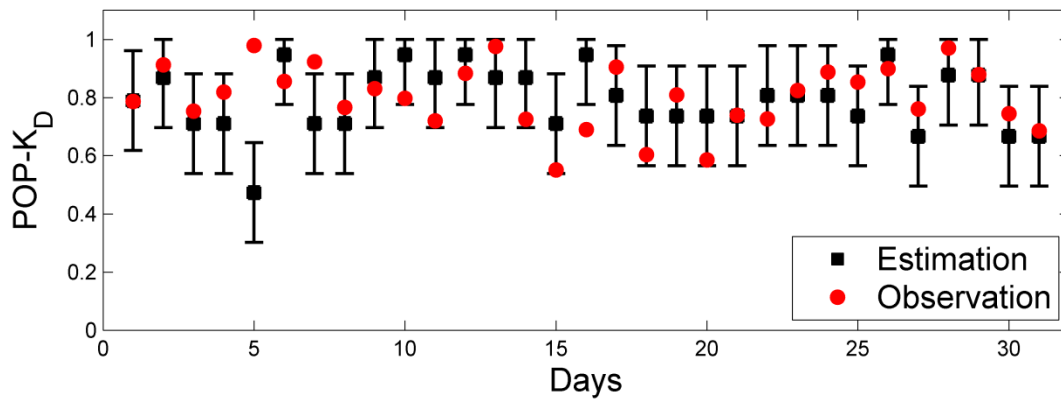


Fig. 5.13: Comparison of the estimated and observed $POP-K_D$ with the 95% confidence intervals (CIs) for the first 31 days of the year 2012.

5.8. Chapter contributions

This chapter provides several contributions. Firstly, this chapter demonstrates that POP- K_D efficiently represents the quality of daily solar irradiance. Higher POP- K_D levels show higher certainty and higher kurtosis (sharper peakedness) in the distribution of the per-minute changes in daily solar irradiance. Thus, the distributional pattern can be represented by a single value of POP- K_D . In addition, POP- K_D indicates the probability that solar irradiance variability is within the ramp rates of common generators in power systems at a PV penetration level. Secondly, this chapter describes the conversion from cloud fluctuation information to quality of surface solar irradiance with a new equation. This conversion equation was developed based on the multi-year relationship between the sky imager-based cloud observation and the measured solar irradiance. The proposed conversion achieves accuracy. Lastly, this chapter proposes a day-ahead solar irradiance quality estimation method using POP- K_D based on fluctuation information provided by the NWS sky cover forecast. This method uses a normalization approach to relate fluctuation of cloud forecast and fluctuation of cloud observation. The observed data are regularly found within the 95% CIs of the estimated values.

CHAPTER 6. CONCLUSION AND FUTURE WORK

6.1. Conclusion

This dissertation introduces the new characterization method (K-POP method) for daily sky conditions using the 2-tuple: K_D and POP- K_D . Based on the K-POP method, this dissertation also introduces the classification method for daily sky conditions. The grid plotted by the K-POP method is divided into ten regions (classes) corresponding to different sky conditions. The both methods are developed by using the multi-year ground-based irradiance measurement data collected at three U.S. stations: SRRL BMS, UNLV, and ORNL.

The quality of daily solar irradiance is analyzed by the newly proposed measure, POP- K_D . During a day, POP- K_D observes the probability of steady irradiance capable of providing smoother electric power to the grid. This quality shows the different ranges at the different quantity levels. The annual distributions of the K-POP method for all three locations are enclosed by the same irregular nonagon boundary. This boundary indicates that the quality is varying at the middle level of the quantity, but high and more certain quality can be expected at very high or low quantity level. Although different shape of the boundary may be revealed at different locations, observing the boundary indicates the unique relationship between the quality and quantity at the location.

The characterization and classification results reveal interesting patterns for the three locations. First, the K-POP method shows significantly annual consistencies of the 2-tuple for each location. The annual mean of the 2-tuple at each location is statistically constant over multiple years. Thus, the 1-year statistics of the classification may represent the multiple-year statistical results. In addition, the seasonal trends for K_D and POP- K_D display similar patterns for

the same location since one seasonal trend for both the 2-tuple together is different or located downwards from the other two seasonal trends for all three locations. The classification result shows the existence of common dominant class (Class 1, 4, and 5), and the transitions between the dominant classes are significant for the three locations. In addition, the monthly observation of the classes indicates different patterns for different locations. Therefore, this monthly observation may be useful to characterize the nature of solar irradiance (quantity and quality) in each month at the location.

The synthetic sequences of the 2-tuple are generated for the three locations based on the Markov approach. For both 2-tuple, the similarities between the synthetic and observed sequences are verified by the statistical comparisons and tests. These generated sequences are considered to be useful for substituting the limited observed data at the location.

This dissertation also proposes new methodologies for estimating the day-ahead solar irradiance quantity from the NWS total sky cover forecast. First of all, the uncertainty for the NWS sky cover-based estimation process was quantified. The previously published method of direct conversion from the NWS total sky cover to solar irradiance was split up into two stages: forecast verification and cloud-to-irradiance conversion. This split-up allows quantification of the uncertainty for each stage and for the overall estimation process. The NWS sky cover forecast shows approximately 20% uncertainty. This amount is identified as the main source of uncertainty for the overall process.

The need to adjust the NWS sky cover forecasts is demonstrated for improving the estimation accuracy. The NWS sky cover forecast was verified by the sky imager-based cloud observation. The verification result indicates that the sky cover forecast is generally about 10%

overestimated overall. The days with high irradiance shows higher overestimation (about 20%), but the bias generally disappears in the days with medium and low irradiance. Therefore, adaptive adjustment at different irradiance levels is recommended for handling this bias in method development.

A new cloud-to-irradiance equation is proposed for the conversion from daily cloud quantity to daily quantity of surface solar irradiance. This equation was developed based on the multi-year relationship between the sky imager-based cloud observation and measured solar irradiance. The proposed equation achieves accuracy with simplicity and its uncertainty can be quantified.

It is proposed five day-ahead quantity estimation methods that apply different adjustment schemes to account for the bias of the sky cover forecast. Method 1 uses zero adjustment. Methods 2, 3, and 4 incorporate the bias ($TKC_{D,bias}$) in Stage 2 by subtracting μ_{adj} . Method 5 incorporates the bias in Stage 3 by partially shifting the TKC_D -vs- K_D curve. Results verify the need for bias adjustment in either Stage 2A or Stage 3A based on the comparisons of Method 1 and the other methods. The results also indicate that the adaptive bias adjustment at different irradiance levels (Methods 3, 4, and 5) is more effective than the single-value adjustment (Method 2). Considering all factors, Method 4 is better than Method 3. Method 5 shows the best performance in terms of KSI and OVER. The observed data are regularly found within the 95% CIs of the forecasted values.

This dissertation also proposes a new methodology for estimating day-ahead solar irradiance quality using the NWS sky cover forecast. Firstly, this chapter demonstrates that POP- K_D efficiently represents daily solar irradiance quality. From the multi-year data, a higher POP-

K_D shows higher certainty and higher kurtosis in the distribution of per-minute changes of daily solar irradiance. This result indicates that a single value of POP- K_D can represent the overall pattern of the distribution. In addition, POP- K_D indicates the probability that the per-minute fluctuation of solar irradiance during a given day is within the ramp rates of the common generators in power systems at a certain PV penetration level.

A new equation for the conversion from cloud fluctuation information to quality of surface solar irradiance is proposed for the day-ahead quality estimation method. This equation was developed based on the multi-year relationship between the sky imager-based cloud observation and measured solar irradiance. The proposed equation achieves accuracy.

It is proposed a day-ahead quality estimation method based on fluctuation information from the NWS sky cover forecast. This method applies normalization by range to relate the fluctuation of the forecast cloud cover to the fluctuation of the observed cloud cover. Results show approximately 16% of RMSE, 0.24% of MBE, 54.83% of KSI, and 3.49% of OVER for the tested year. The observed POP- K_D data are regularly found within the 95% CIs of the estimated values.

6.2. Contribution

Overall, the major contributions of this dissertation are summarized as follows.

- A new parameter, POP-K_D , has been proposed for characterizing quality of daily solar irradiance. POP-K_D efficiently represents daily solar irradiance quality because a single value of POP-K_D can represent the overall pattern of the daily solar irradiance fluctuation distribution. In addition, POP-K_D can be applicable to indicates that the per-minute fluctuation of solar irradiance during a given day is within the ramp rates of the common generators in power systems at a certain PV penetration level.
- This dissertation has proposed a new characterization method for daily sky conditions by using the 2-tuple (K_D and POP-K_D). The annual distribution of the proposed method relates the quality of irradiance to the quantity. In addition, an annual statistical result may be representative of multiple-year statistical results at a certain location.
- This dissertation has proposed a new classification method for daily sky conditions. The grid plotted by the proposed characterization method is divided into ten regions (classes) corresponding to different sky conditions. Several dominant classes exist commonly, and transitions among the dominant classes are significantly frequent compared to transitions among other classes. Monthly patterns of the dominant classes at a location can be useful to characterize the local nature of solar irradiance.
- This dissertation has demonstrated the process to generate synthetic sequences of K_D and POP-K_D based on a first-order Markov approach. The synthetic sequences can be useful to optimize the solar-based generation system and substitute limited measured data.
- This dissertation has quantified the uncertainty for the estimation process from the NWS sky cover forecast to the solar irradiance quantity. The NWS sky cover forecast shows

approximately 20% uncertainty. This amount is identified as the main source of uncertainty for the overall process.

- This dissertation has demonstrated the need to adjust the NWS sky cover forecasts. Verification of the sky cover forecast reveals an overall overestimated bias. This bias is stronger at high irradiance levels. Thus, the estimation method needs to adaptively handle this bias at different irradiance levels.
- A retrieval process for the historical total sky cover from NDFD has been described in detail. In addition, processing of the retrieved sky cover sequences has been demonstrated for achieving the day-ahead average/POP of the sky cover forecast.
- This dissertation has described the conversions from daily cloud quantity/quality to surface solar irradiance quantity/quality with new equations. The proposed conversions achieve accuracy with simplicity.
- This dissertation has introduced five improved methods to estimate day-ahead K_D value using the NWS total sky cover forecast. These methods incorporate different schemes for dealing with the aforementioned bias of the cloud forecast. In addition, this dissertation has proposed a new method to estimate day-ahead POP- K_D value using cloud fluctuation information provided by the NWS sky cover forecast.

6.3. Future works

The following future works are suggested to improve the feasibility of the proposed characterization, classification, and estimation methods.

Firstly, more data will be utilized to enhance the library of transition matrices of K_D and POP- K_D . More studies will be conducted to further benchmark the synthetic sequences. In addition, the suggested ten classes of the new classification method for daily sky conditions can be applied to more sophisticated statistic-based prediction techniques.

Secondly, the proposed estimation methods for day-ahead solar irradiance quantity and quality were developed by the data collected at a particular location, SRRL BMS. Publicly available data measured by ground-based total sky imagers with high temporal resolution are significantly limited. Thus, more spatially diverse data in future studies would improve the generalizability of the proposed estimation methods.

Lastly, the proposed methods use daily values of solar irradiance quantity and quality. Solar industry also favors accurate characterization and prediction results including more specified time horizons (e.g. hourly value of solar irradiance). Thus, it would be valuable to characterize, classify, and estimate sky conditions with the reduced prediction time horizon based on modification of the proposed methods in future studies.

APPENDIX A. MATLAB[®] SOURCE CODE

A.1. Description of MAT-files

A.1.1. *Golden_2001_2011_YearlyGroup_idx_perK_proS.mat*

- This file contains annual groups of K_D , POP- K_D , and Class at SRRL BMS (Golden, CO) from 2001 to 2011.
- The variable of “perK_Golden_[year]” indicates the K_D values at SRRL BMS and the specific year.
- The variable of “proS_Golden_[year]” indicates the POP- K_D values at SRRL BMS and the specific year.
- The variable of “idx_Golden_[year]” indicates the classified class values at SRRL BMS and the specific year.

A.1.2. *UNLV_2007_2011_YearlyGroup_idx_perK_proS.mat*

- This file contains annual groups of K_D , POP- K_D , and Class at UNLV (Las Vegas, NV) from 2007 to 2011.
- The variable of “perK_UNLV_[year]” indicates the K_D values at UNLV and the specific year.
- The variable of “proS_UNLV_[year]” indicates the POP- K_D values at UNLV and the specific year.
- The variable of “idx_UNLV_[year]” indicates the classified class values at UNLV and the specific year.

A.1.3. ORNL_2008_2011_YearlyGroup_idx_perK_proS.mat

- This file contains annual groups of K_D , POP- K_D , and Class at ORNL (Oak Ridge, TN) from 2008 to 2011.
- The variable of “perK_ORNL_[year]” indicates the K_D values at ORNL and the specific year.
- The variable of “proS_ORNL_[year]” indicates the POP- K_D values at ORNL and the specific year.
- The variable of “idx_ORNL_[year]” indicates the classified class values at ORNL and the specific year.

A.1.4. AugiarMTM_All_Locations_Monthly_perK_proS_classes.mat

- This file contains the library of Markov Transition Matrices (MTMs) for K_D and POP- K_D , and these values are used for the synthetic sequence generation.
- The structure of “MTMperK” indicates the the library of MTMs for K_D from Class 1 to Class 10.
- The structure of “MTMproS” indicates the the library of MTMs for POP- K_D from Class 1 to Class 10.

A.1.5. Golden_2009_2012_YearlyGroup_ClassGroup_TKCDfor_TKCDobs.mat

- This files contains annual groups of K_D , TKC_D , and $TKC_{D,NWS}$ at SRRL BMS from 2009 to 2012.
- Several days were missing in the forecast data. Thus, these days were also dropped from the observed data at each year.

- The variable of “TCCDfor_[year]” indicates $TKC_{D,NWS}$ at the specific year.
- The variable of “TCCDobs_[year]” indicates TKC_D at the specific year.
- The variable of “[TCCDfor/TCCDobs]_[year]_cls147” indicates $TKC_{D,NWS}$ or TKC_D at the specific year and at the high K_D level.
- The variable of “[TCCDfor/TCCDobs]_[year]_cls258” indicates $TKC_{D,NWS}$ or TKC_D at the specific year and at the medium K_D level.
- The variable of “[TCCDfor/TCCDobs]_[year]_cls369” indicates $TKC_{D,NWS}$ or TKC_D at the specific year and at the low K_D level.

A.1.6. Golden_2001_2012_YearlyGroup_KD_POPKD_Class_stdQD_kurto.mat

- This files contains annual groups of K_D , POP- K_D , and standard deviation and kurtosis of daily probability density function (PDF) of $\Delta\tilde{k}$ at SRRL BMS from 2001 to 2012.
- These values are used to observe the 3-D synoptic view of K_D , POP- K_D , and standard deviation/kurtosis.
- The variable of “stdQD_Golden_[year]” indicates the standard deviation of daily PDF of $\Delta\tilde{k}$ at the specific year.
- The variable of “kurto_Golden_[year]” indicates the kurtosis of daily PDF of $\Delta\tilde{k}$ at the specific year.

A.1.7. Golden_2009_2012_YearlyGroup_POPTKCDfor_POPTKCDobs.mat

- This files contains annual groups of POP- TKC_D and POP- $TKC_{D,NWS}$ at SRRL BMS from 2009 to 2012.

- Several days were missing in the forecast data. Thus, these days were also dropped from the observed data at each year.
- The variable of “POPTKCDfor_[year]” indicates POP-TKC_{D,NWS} at the specific year.
- The variable of “POPTKCDobs_[year]” indicates POP-TKC_D at the specific year.

A.1.8. Golden_2001_2011_YearlyGroup_MonthlyGroup_GHI.mat

- This file contains annual and monthly groups of I_{GHI} at SRRL BMS from 2001 to 2011.
- These values are used for the ramp rate analysis in Section 5.3.3.
- The variable of “GHI_Golden_[year]” indicates the per-minute I_{GHI} at the specific year.
- The variable of “GHI_Golden_[month]_2001_2011” indicates the per-minute I_{GHI} at the specific month for 11 years (2001 – 2011).

A.1.9. Golden_2006_2012_YearlyGroup_TKCD_OKCD_POPTKCD_POPOKCD.mat

- This file contains annual groups of TKC_D, OKC_D (daily quantity of opaque sky cover), POP-TKC_D, and POP-OKC_D (daily POP value of opaque sky cover) calculated by the ground-based sky imager at SRRL BMS from 2009 to 2012.
- All annual data consist of 365 days.
- The variable of “TCC_Golden_[year]” indicates TKC_D at the specific year.
- The variable of “OCC_Golden_[year]” indicates OKC_D at the specific year.
- The variable of “POPTCC_Golden_[year]” indicates POP-TKC_D at the specific year.
- The variable of “POPOCC_Golden_[year]” indicates POP-OKC_D at the specific year.

A.1.10. Golden_2012_YearlyGroup_KD_POPKD_TKCD_POPTKCD.mat

- This file contains annual groups of K_D , $POP\text{-}K_D$, TKC_D , $TKC_{D,NWS}$, $POP\text{-}TKC_D$, and $POP\text{-}TKC_{D,NWS}$ at SRRL BMS for the year of 2012.
- The variable of “KDobs_2012” indicates K_D for the year of 2012.
- The variable of “POPKDobs_2012” indicates $POP\text{-}K_D$ for the year of 2012.
- The variable of “TKCDobs_2012” indicates TKC_D for the year of 2012.
- The variable of “TKCDfor_2012” indicates $TKC_{D,NWS}$ for the year of 2012.
- The variable of “POPTKCDobs_2012” indicates $POP\text{-}TKC_D$ for the year of 2012.
- The variable of “POPTKCDfor_2012” indicates $POP\text{-}TKC_{D,NWS}$ for the year of 2012.

A.2. M-files for Chapter 3

A.2.1. Annual characterization results at SRRL BMS, UNLV, and ORNL (Section 3.4.1)

● AnnualCharacterizationResult_SRRLBMS.m

```
%% Annual Characterization Result (SRRL BMS)
% This script plots 2-D annual characterization result at SRRL BMS from
% 2001 to 2011.

clear all; close all; clc;

load Golden_2001_2011_YearlyGroup_idx_perK_proS.mat

%% Center of annual distribution
begYr = 2001; % The beginning year
endYr = 2011; % The ending year

selYr = endYr - begYr + 1; % selected years

idx_Golden_Mnly = zeros(12,31*selYr);

center_perK = zeros(selYr+1,1);
center_proS = zeros(selYr+1,1);

for i = 1:selYr
    refYr = begYr + (i-1);
    yrIdx_perK = eval(strcat('perK_Golden_',num2str(refYr)));
    yrIdx_proS = eval(strcat('proS_Golden_',num2str(refYr)));
    center_perK(i) = mean(yrIdx_perK);
    center_proS(i) = mean(yrIdx_proS);
end

center_perK(selYr+1) = mean(center_perK(1:selYr));
center_proS(selYr+1) = mean(center_proS(1:selYr));

%% Plot
% Points for the nonagon boundary
pt1 = [0.8350 0.9900];
pt2 = [0.8132 0.6719];
pt3 = [0.6299 0.5065];
pt4 = [0.4387 0.5158];
pt5 = [0.0541 0.8462];
pt6 = [0.0264 0.9564];
pt7 = [0.1006 0.9804];
pt8 = [0.5004 0.9094];
pt9 = [0.6875 0.9768];

line12_x = linspace(pt1(1),pt2(1)); line12_y = linspace(pt1(2),pt2(2));
line23_x = linspace(pt2(1),pt3(1)); line23_y = linspace(pt2(2),pt3(2));
line34_x = linspace(pt3(1),pt4(1)); line34_y = linspace(pt3(2),pt4(2));
```

```

line45_x = linspace(pt4(1),pt5(1)); line45_y = linspace(pt4(2),pt5(2));
line56_x = linspace(pt5(1),pt6(1)); line56_y = linspace(pt5(2),pt6(2));
line67_x = linspace(pt6(1),pt7(1)); line67_y = linspace(pt6(2),pt7(2));
line78_x = linspace(pt7(1),pt8(1)); line78_y = linspace(pt7(2),pt8(2));
line89_x = linspace(pt8(1),pt9(1)); line89_y = linspace(pt8(2),pt9(2));
line91_x = linspace(pt9(1),pt1(1)); line91_y = linspace(pt9(2),pt1(2));

% Size and width
liWidth = 1; % line width
mSize = 10; % Marker size
cmSize = 8; % Center marker size
liRe = 10; % Line resolution
nPts = 50;

%% SRRL BMS 2001
f1 = figure(1);
subplot(221)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_2001,proS_Golden_2001,'r.','MarkerSize',mSize)
plot(center_perK(1),center_proS(1),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(a) SRRL BMS, 2001')
xlim([0 0.9]);ylim([0.4 1])

%% SRRL BMS 2002
subplot(222)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_2002,proS_Golden_2002,'r.','MarkerSize',mSize)
plot(center_perK(2),center_proS(2),'o','MarkerEdgeColor','k',...

```

```

    'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(b) SRRL BMS, 2002')
xlim([0 0.9]);ylim([0.4 1])

%%% SRRL BMS 2003
subplot(223)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_2003,proS_Golden_2003,'r.','MarkerSize',mSize)
plot(center_perK(3),center_proS(3),'o','MarkerEdgeColor','k',...
    'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(c) SRRL BMS, 2003')
xlim([0 0.9]);ylim([0.4 1])

%%% SRRL BMS 2004
subplot(224)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

```

```

plot(perK_Golden_2004,proS_Golden_2004,'r.','MarkerSize',mSize)
plot(center_perK(4),center_proS(4),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(d) SRRL BMS, 2004')
xlim([0 0.9]);ylim([0.4 1])

%print(f1,'-dtiff','-r500','Golden_Dist(1)');

%%% SRRL BMS 2005
f2 = figure(2);
subplot(221)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_2005,proS_Golden_2005,'r.','MarkerSize',mSize)
plot(center_perK(5),center_proS(5),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(e) SRRL BMS, 2005')
xlim([0 0.9]);ylim([0.4 1])

%%% SRRL BMS 2006
subplot(222)

```

```

plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_2006,proS_Golden_2006,'r.','MarkerSize',mSize)
plot(center_perK(6),center_proS(6),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(f) SRRL BMS, 2006')
xlim([0 0.9]);ylim([0.4 1])

%%% SRRL BMS 2007
subplot(223)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_2007,proS_Golden_2007,'r.','MarkerSize',mSize)
plot(center_perK(7),center_proS(7),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(g) SRRL BMS, 2007')
xlim([0 0.9]);ylim([0.4 1])

```



```

%%% SRRL BMS 2008
subplot(224)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_2008,proS_Golden_2008,'r.','MarkerSize',mSize)
plot(center_perK(8),center_proS(8),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(h) SRRL BMS, 2008')
xlim([0 0.9]);ylim([0.4 1])

%print(f2,'-dtiff','-r500','Golden_Dist(2)');

%%% SRRL BMS 2009
f3 = figure(3);
subplot(221)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_2009,proS_Golden_2009,'r.','MarkerSize',mSize)
plot(center_perK(9),center_proS(9),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)

```

```

plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(i) SRRL BMS, 2009')
xlim([0 0.9]);ylim([0.4 1])

%%% SRRL BMS 2010
subplot(222)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_2010,proS_Golden_2010,'r.','MarkerSize',mSize)
plot(center_perK(10),center_proS(10),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(j) SRRL BMS, 2010')
xlim([0 0.9]);ylim([0.4 1])

%%% SRRL BMS 2011
subplot(223)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_2011,proS_Golden_2011,'r.','MarkerSize',mSize)
plot(center_perK(11),center_proS(11),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)

```

```

plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(k) SRRL BMS, 2011')
xlim([0 0.9]);ylim([0.4 1])

%%% SRRL BMS 2001 ~ 2011
subplot(224)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_Golden_alYr,proS_Golden_alYr,'r.','MarkerSize',mSize)
plot(center_perK(12),center_proS(12),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)

% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(l) SRRL BMS, 2001 ~ 2011')
xlim([0 0.9]);ylim([0.4 1])

%print(f3,'-dtiff','-r500','Golden_Dist(3)');

```

● AnnualCharacterizationResult_UNLV_ORNL.m

```

%% Annual Characterization Result (UNLV and ORNL)
% This script plots 2-D annual characterization result at UNLV from 2007
% to 2011 and at ORNL from 2008 to 2011.

clear all; close all; clc;

load UNLV_2007_2011_YearlyGroup_idx_perK_proS.mat
load ORNL_2008_2011_YearlyGroup_idx_perK_proS.mat

%% Center of annual distribution (UNLV)

```

```

begYr = 2007; % The beginning year
endYr = 2011; % The ending year

selYr = endYr - begYr + 1; % selected years

center_UNLV_perK = zeros(selYr+1,1);
center_UNLV_proS = zeros(selYr+1,1);

for i = 1:selYr
    refYr = begYr + (i-1);
    yrIdx_perK = eval(strcat('perK_UNLV_',num2str(refYr)));
    yrIdx_proS = eval(strcat('proS_UNLV_',num2str(refYr)));
    center_UNLV_perK(i) = mean(yrIdx_perK);
    center_UNLV_proS(i) = mean(yrIdx_proS);
end

center_UNLV_perK(selYr+1) = mean(center_UNLV_perK(1:selYr));
center_UNLV_proS(selYr+1) = mean(center_UNLV_proS(1:selYr));

%% Center of annual distribution (ORNL)
begYr = 2008; % The beginning year
endYr = 2011; % The ending year

selYr = endYr - begYr + 1; % selected years

center_ORNL_perK = zeros(selYr+1,1);
center_ORNL_proS = zeros(selYr+1,1);

for i = 1:selYr
    refYr = begYr + (i-1);
    yrIdx_perK = eval(strcat('perK_ORNL_',num2str(refYr)));
    yrIdx_proS = eval(strcat('proS_ORNL_',num2str(refYr)));
    center_ORNL_perK(i) = mean(yrIdx_perK);
    center_ORNL_proS(i) = mean(yrIdx_proS);
end

center_ORNL_perK(selYr+1) = mean(center_ORNL_perK(1:selYr));
center_ORNL_proS(selYr+1) = mean(center_ORNL_proS(1:selYr));
%% Setup points and lines
% Points for the nonagon boundary
pt1 = [0.8350 0.9900];
pt2 = [0.8132 0.6719];
pt3 = [0.6299 0.5065];
pt4 = [0.4387 0.5158];
pt5 = [0.0541 0.8462];
pt6 = [0.0264 0.9564];
pt7 = [0.1006 0.9804];
pt8 = [0.5004 0.9094];
pt9 = [0.6875 0.9768];

line12_x = linspace(pt1(1),pt2(1)); line12_y = linspace(pt1(2),pt2(2));
line23_x = linspace(pt2(1),pt3(1)); line23_y = linspace(pt2(2),pt3(2));
line34_x = linspace(pt3(1),pt4(1)); line34_y = linspace(pt3(2),pt4(2));
line45_x = linspace(pt4(1),pt5(1)); line45_y = linspace(pt4(2),pt5(2));
line56_x = linspace(pt5(1),pt6(1)); line56_y = linspace(pt5(2),pt6(2));

```

```

line67_x = linspace(pt6(1),pt7(1)); line67_y = linspace(pt6(2),pt7(2));
line78_x = linspace(pt7(1),pt8(1)); line78_y = linspace(pt7(2),pt8(2));
line89_x = linspace(pt8(1),pt9(1)); line89_y = linspace(pt8(2),pt9(2));
line91_x = linspace(pt9(1),pt1(1)); line91_y = linspace(pt9(2),pt1(2));

% Size and width
liWidth = 1; % line width
mSize = 10; % Marker size
cmSize = 8; % Center marker size
nPts = 50;

%% UNLV Plot

%%% UNLV 2007
f1 = figure(1);
subplot(221)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_UNLV_2007,proS_UNLV_2007,'r.','MarkerSize',mSize)
plot(center_UNLV_perK(3),center_UNLV_proS(3),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(a) UNLV, 2007')
xlim([0 0.9]);ylim([0.4 1])

%%% UNLV 2008
subplot(222)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_UNLV_2008,proS_UNLV_2008,'r.','MarkerSize',mSize)
plot(center_UNLV_perK(3),center_UNLV_proS(3),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary

```

```

plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(b) UNLV, 2008')
xlim([0 0.9]);ylim([0.4 1])

%%% UNLV 2009
subplot(223)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_UNLV_2009,proS_UNLV_2009,'r.','MarkerSize',mSize)
plot(center_UNLV_perK(3),center_UNLV_proS(3),'o','MarkerEdgeColor','k',...
'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(c) UNLV, 2009')
xlim([0 0.9]);ylim([0.4 1])

%%% UNLV 2010
subplot(224)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_UNLV_2010,proS_UNLV_2010,'r.','MarkerSize',mSize)
plot(center_UNLV_perK(4),center_UNLV_proS(4),'o','MarkerEdgeColor','k',...
'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary

```

```

plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)
hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(d) UNLV, 2010')
xlim([0 0.9]);ylim([0.4 1])

%print(f1,'-dtiff','-r500','UNLV_Dist(1)');

%%% UNLV 2011
f2 = figure(2);
subplot(221)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_UNLV_2011,proS_UNLV_2011,'r.','MarkerSize',mSize)
plot(center_UNLV_perK(5),center_UNLV_proS(5),'o','MarkerEdgeColor','k',...
'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(e) UNLV, 2011')
xlim([0 0.9]);ylim([0.4 1])

%%% UNLV 2009 ~ 2011
subplot(222)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_UNLV_alYr,proS_UNLV_alYr,'r.','MarkerSize',mSize)
plot(center_UNLV_perK(6),center_UNLV_proS(6),'o','MarkerEdgeColor','k',...

```

```

    'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(f) UNLV, 2007 ~ 2011')
xlim([0 0.9]);ylim([0.4 1])

%print(f2,'-dtiff','-r500','UNLV_Dist(2)');

%% ORNL Plot
%%% ORNL 2008
f3 = figure(3);
subplot(221)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_ORNL_2008,proS_ORNL_2008,'r.','MarkerSize',mSize)
plot(center_ORNL_perK(2),center_ORNL_proS(2),'o','MarkerEdgeColor','k',...
    'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(a) ORNL, 2008')
xlim([0 0.9]);ylim([0.4 1])

%%% ORNL 2009
subplot(222)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')

```



```

plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_ORNL_2009,proS_ORNL_2009,'r.','MarkerSize',mSize)
plot(center_ORNL_perK(2),center_ORNL_proS(2),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(b) ORNL, 2009')
xlim([0 0.9]);ylim([0.4 1])

%%% ORNL 2010
subplot(223)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_ORNL_2010,proS_ORNL_2010,'r.','MarkerSize',mSize)
plot(center_ORNL_perK(3),center_ORNL_proS(3),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(c) ORNL, 2010')
xlim([0 0.9]);ylim([0.4 1])

%%% ORNL 2011
subplot(224)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')

```

```

plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_ORNL_2011,proS_ORNL_2011,'r.','MarkerSize',mSize)
plot(center_ORNL_perK(4),center_ORNL_proS(4),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(d) ORNL, 2011')
xlim([0 0.9]);ylim([0.4 1])

%print(f3,'-dtiff','-r500','ORNL_Dist(1)');

%% ORNL 2009 ~ 2011
f4 = figure(4);
subplot(221)
plot(linspace(0,1,nPts),linspace(0.9,0.9,nPts),'k-.')
hold on
plot(linspace(0,1,nPts),linspace(0.7,0.7,nPts),'k-.')
plot(linspace(0,1,nPts),linspace(0.5,0.5,nPts),'k-.')
plot(linspace(0.6,0.6,nPts),linspace(0.5,1,nPts),'k-.')
plot(linspace(0.3,0.3,nPts),linspace(0.5,1,nPts),'k-.')

plot(perK_ORNL_alYr,proS_ORNL_alYr,'r.','MarkerSize',mSize)
plot(center_ORNL_perK(5),center_ORNL_proS(5),'o','MarkerEdgeColor','k',...
      'MarkerFaceColor','k','MarkerSize',cmSize)
% Nonagon boundary
plot(line12_x,line12_y,'k-','LineWidth',liWidth)
plot(line23_x,line23_y,'k-','LineWidth',liWidth)
plot(line34_x,line34_y,'k-','LineWidth',liWidth)
plot(line45_x,line45_y,'k-','LineWidth',liWidth)
plot(line56_x,line56_y,'k-','LineWidth',liWidth)
plot(line67_x,line67_y,'k-','LineWidth',liWidth)
plot(line78_x,line78_y,'k-','LineWidth',liWidth)
plot(line89_x,line89_y,'k-','LineWidth',liWidth)
plot(line91_x,line91_y,'k-','LineWidth',liWidth)

hold off
xlabel('Daily Sky Clearness Index'); ylabel('Probability of Persistence');
title('(e) ORNL, 2008 ~ 2011')
xlim([0 0.9]);ylim([0.4 1])

%print(f4,'-dtiff','-r500','ORNL_Dist(2)');

```

A.2.2. Annual means of K_D and POP- K_D at SRRL BMS, UNLV, and ORNL (Section 3.4.1)

● Errorbar_AnnualMean_KD_POPKD_SRRLBMS_UNLV_ORNL.m

```
%% Errorbars of Annual Mean for KD and POPKD at All Locations
% This script plots errorbars of annual mean +- standard deviation for KD
% and POP-KD at SRRL BMS, UNLV, and ORNL.

clear all; close all; clc;

load Golden_2001_2011_YearlyGroup_idx_perK_proS.mat
load UNLV_2007_2011_YearlyGroup_idx_perK_proS.mat
load ORNL_2008_2011_YearlyGroup_idx_perK_proS.mat

%% Centers
% SRRL BMS
loca = 'Golden';

begYr = 2001;
endYr = 2011;
selYr = endYr - begYr + 1;

yrAvePerK_temp = zeros(1,selYr);
yrStdPerK_temp = zeros(1,selYr);

yrAveProS_temp = zeros(1,selYr);
yrStdProS_temp = zeros(1,selYr);

for i = 1:selYr
    refYr = begYr + (i-1);
    yrPerK_temp = eval(strcat('perK_',loca,'_',num2str(refYr)));
    yrAvePerK_temp(i) = mean(yrPerK_temp);
    yrStdPerK_temp(i) = std(yrPerK_temp);

    yrProS_temp = eval(strcat('proS_',loca,'_',num2str(refYr)));
    yrAveProS_temp(i) = mean(yrProS_temp);
    yrStdProS_temp(i) = std(yrProS_temp);
end

yrAvePerK_Golden = yrAvePerK_temp; yrStdPerK_Golden = yrStdPerK_temp;
yrAveProS_Golden = yrAveProS_temp; yrStdProS_Golden = yrStdProS_temp;
yrs_Golden = [begYr:endYr];

% UNLV
loca = 'UNLV';

begYr = 2007;
endYr = 2011;
selYr = endYr - begYr + 1;

yrAvePerK_temp = zeros(1,selYr);
yrStdPerK_temp = zeros(1,selYr);
```

```

yrAveProS_temp = zeros(1,selYr);
yrStdProS_temp = zeros(1,selYr);

for i = 1:selYr
    refYr = begYr + (i-1);
    yrPerK_temp = eval(strcat('perK_',loca,'_',num2str(refYr)));
    yrAvePerK_temp(i) = mean(yrPerK_temp);
    yrStdPerK_temp(i) = std(yrPerK_temp);

    yrProS_temp = eval(strcat('proS_',loca,'_',num2str(refYr)));
    yrAveProS_temp(i) = mean(yrProS_temp);
    yrStdProS_temp(i) = std(yrProS_temp);
end

yrAvePerK_UNLV = yrAvePerK_temp; yrStdPerK_UNLV = yrStdPerK_temp;
yrAveProS_UNLV = yrAveProS_temp; yrStdProS_UNLV = yrStdProS_temp;
yrs_UNLV = [begYr:endYr];

% ORNL
loca = 'ORNL';

begYr = 2008;
endYr = 2011;
selYr = endYr - begYr + 1;

yrAvePerK_temp = zeros(1,selYr);
yrStdPerK_temp = zeros(1,selYr);

yrAveProS_temp = zeros(1,selYr);
yrStdProS_temp = zeros(1,selYr);

for i = 1:selYr
    refYr = begYr + (i-1);
    yrPerK_temp = eval(strcat('perK_',loca,'_',num2str(refYr)));
    yrAvePerK_temp(i) = mean(yrPerK_temp);
    yrStdPerK_temp(i) = std(yrPerK_temp);

    yrProS_temp = eval(strcat('proS_',loca,'_',num2str(refYr)));
    yrAveProS_temp(i) = mean(yrProS_temp);
    yrStdProS_temp(i) = std(yrProS_temp);
end

yrAvePerK_ORNL = yrAvePerK_temp; yrStdPerK_ORNL = yrStdPerK_temp;
yrAveProS_ORNL = yrAveProS_temp; yrStdProS_ORNL = yrStdProS_temp;
yrs_ORNL = [begYr:endYr];

%% Plot
lnWidth = 1.5;
mkSize = 5;
loAdj = 0.15;

f1 = figure(1);
subplot(211)

```

```

errorbar(yrs_Golden, yrAvePerK_Golden, yrStdPerK_Golden, 'k-o', ...
         'LineWidth', lnWidth, 'MarkerFaceColor', 'k', 'MarkerSize', mkSize)
hold on
errorbar(yrs_UNLV-loAdj, yrAvePerK_UNLV, yrStdPerK_UNLV, 'r-o', ...
         'LineWidth', lnWidth, 'MarkerFaceColor', 'r', 'MarkerSize', mkSize)
errorbar(yrs_ORNL+loAdj, yrAvePerK_ORNL, yrStdPerK_ORNL, 'b-o', ...
         'LineWidth', lnWidth, 'MarkerFaceColor', 'b', 'MarkerSize', mkSize)
% plot(yrs_Golden, yrAvePerK_Golden, 'ko', ...
%      'LineWidth', lnWidth, 'MarkerFaceColor', 'k', 'MarkerSize', mkSize)
hold off
xlim([2000.5 2011.5]); ylim([0 1]);
ylabel('K_D'); xlabel('Year');
legend('SRRL BMS', 'UNLV', 'ORNL', 'Location', 'SouthWest')
title('(a)');

subplot(212)
errorbar(yrs_Golden, yrAveProS_Golden, yrStdProS_Golden, 'k-o', ...
         'LineWidth', lnWidth, 'MarkerFaceColor', 'k', 'MarkerSize', mkSize)
hold on
errorbar(yrs_UNLV-loAdj, yrAveProS_UNLV, yrStdProS_UNLV, 'r-o', ...
         'LineWidth', lnWidth, 'MarkerFaceColor', 'r', 'MarkerSize', mkSize)
errorbar(yrs_ORNL+loAdj, yrAveProS_ORNL, yrStdProS_ORNL, 'b-o', ...
         'LineWidth', lnWidth, 'MarkerFaceColor', 'b', 'MarkerSize', mkSize)
plot(yrs_Golden, yrAveProS_Golden, 'ko', ...
     'LineWidth', lnWidth, 'MarkerFaceColor', 'k', 'MarkerSize', mkSize)
hold off
xlim([2000.5 2011.5]); ylim([0.5 1]);
ylabel('POP-K_D'); xlabel('Year');
legend('SRRL BMS', 'UNLV', 'ORNL', 'Location', 'SouthWest')
title('(b)');

```

A.2.3. MTMs and monthly patterns at SRRL BMS, UNLV, and ORNL (Section 3.4.2 & 3.4.3)

● MTMs_MonthlyDominantClasses_SRRLBMS_UNLV_ORNL.m

```

%% Markov Transition Matrices and Monthly Pattern of Dominant Classes
% This script generates transition matrices based on 10 classes by the
% proposed classification method and calculates monthly patterns of the
% dominant classes.

% Locations: Golden (2001 ~ 2011)
%             UNLV (2007 ~ 2011)
%             ORNL (2008 ~ 2011)

clear all; close all; clc;

load Golden_2001_2011_YearlyGroup_idx_perK_proS.mat
load UNLV_2007_2011_YearlyGroup_idx_perK_proS.mat
load ORNL_2008_2011_YearlyGroup_idx_perK_proS.mat

%% Transition matrix for all years
% SRRL BMS

```

```

idx_Golden = idx_Golden_alYr;
pMat_Golden = genProMatRectClass(idx_Golden);

% UNLV
idx_UNLV = idx_UNLV_alYr;
pMat_UNLV = genProMatRectClass(idx_UNLV);

% ORNL
idx_ORNL = idx_ORNL_alYr;
pMat_ORNL = genProMatRectClass(idx_ORNL);

%% Monthly patterns of dominant classes
nDysPerMn = [31 28 31 30 31 30 31 31 30 31 30 31];

%%% SRRL BMS
begYr = 2001; % The beginning year
endYr = 2011; % The ending year

selYr = endYr - begYr + 1; % selected years

idx_Golden_Mnly = zeros(12,31*selYr);

for i = 1:selYr
    refYr = begYr + (i-1);
    yrIdx = eval(strcat('idx_Golden_',num2str(refYr)));

    preDySum = 0;
    for j = 1:12
        idx_Golden_Mnly(j,(i-1)*nDysPerMn(j)+1:i*nDysPerMn(j))=yrIdx(preDySum+1:preDySum+nDysPerMn(j));
        preDySum = preDySum + nDysPerMn(j);
    end
end

mnDist_Golden = zeros(10,12);
for i = 1:12
    for j = 1:nDysPerMn(i)*selYr
        idxValue = idx_Golden_Mnly(i,j);
        mnDist_Golden(idxValue,i) = mnDist_Golden(idxValue,i) + 1;
    end
    mnDist_Golden(:,i) = mnDist_Golden(:,i) / (nDysPerMn(i)*selYr);
end

%%% UNLV
begYr = 2007; % The beginning year
endYr = 2011; % The ending year

selYr = endYr - begYr + 1; % selected years

idx_UNLV_Mnly = zeros(12,31*selYr);

for i = 1:selYr
    refYr = begYr + (i-1);
    yrIdx = eval(strcat('idx_UNLV_',num2str(refYr)));

```

```

        preDySum = 0;
        for j = 1:12
            idx_UNLV_Mnly(j, (i-
1)*nDysPerMn(j)+1:i*nDysPerMn(j))=yrIdx(preDySum+1:preDySum+nDysPerMn(j));
            preDySum = preDySum + nDysPerMn(j);
        end
    end

mnDist_UNLV = zeros(10,12);
for i = 1:12
    for j = 1:nDysPerMn(i)*selYr
        idxValue = idx_UNLV_Mnly(i,j);
        mnDist_UNLV(idxValue,i) = mnDist_UNLV(idxValue,i) + 1;
    end
    mnDist_UNLV(:,i) = mnDist_UNLV(:,i) / (nDysPerMn(i)*selYr);
end

%%% ORNL
begYr = 2008; % The beginning year
endYr = 2011; % The ending year

selYr = endYr - begYr + 1; % selected years

idx_ORNL_Mnly = zeros(12,31*selYr);

for i = 1:selYr
    refYr = begYr + (i-1);
    yrIdx = eval(strcat('idx_ORNL_',num2str(refYr)));

    preDySum = 0;
    for j = 1:12
        idx_ORNL_Mnly(j, (i-
1)*nDysPerMn(j)+1:i*nDysPerMn(j))=yrIdx(preDySum+1:preDySum+nDysPerMn(j));
        preDySum = preDySum + nDysPerMn(j);
    end
end

mnDist_ORNL = zeros(10,12);
for i = 1:12
    for j = 1:nDysPerMn(i)*selYr
        idxValue = idx_ORNL_Mnly(i,j);
        mnDist_ORNL(idxValue,i) = mnDist_ORNL(idxValue,i) + 1;
    end
    mnDist_ORNL(:,i) = mnDist_ORNL(:,i) / (nDysPerMn(i)*selYr);
end

%%% Plotting
% SRRL BMS
f1 = figure(1);
plot([1:12],mnDist_Golden(1,:)*100,'r-*) % Class 1
hold on
plot([1:12],mnDist_Golden(4,:)*100,'k-o') % Class 4
plot([1:12],mnDist_Golden(5,:)*100,'b-s') % Class 5
plot([1:12],mnDist_Golden(8,:)*100,'m-+') % Class 8

```

```

hold off
xlim([1 12]); xlabel('Month');
ylim([0 50]); ylabel('Percentage (%)');
legend('Class 1','Class 4','Class 5','Class 8')
title('SRRL BMS')
%print(f1,'-dtiff','-r500','Monthly_Class_Pattern_Golden');

% UNLV
f2 = figure(2);
plot([1:12],mnDist_UNLV(1,:)*100,'r-*) % Class 1
hold on
plot([1:12],mnDist_UNLV(4,:)*100,'k-o') % Class 4
plot([1:12],mnDist_UNLV(5,:)*100,'b-s') % Class 5
hold off
xlim([1 12]); xlabel('Month');
ylim([0 70]); ylabel('Percentage (%)');
legend('Class 1','Class 4','Class 5')
title('UNLV')
%print(f2,'-dtiff','-r500','Monthly_Class_Pattern_UNLV');

% ORNL
f3 = figure(3);
plot([1:12],mnDist_ORNL(1,:)*100,'r-*) % Class 1
hold on
plot([1:12],mnDist_ORNL(4,:)*100,'k-o') % Class 4
plot([1:12],mnDist_ORNL(5,:)*100,'b-s') % Class 5
plot([1:12],mnDist_ORNL(6,:)*100,'g-d') % Class 6
plot([1:12],mnDist_ORNL(8,:)*100,'m-+') % Class 8
hold off
xlim([1 12]); xlabel('Month');
ylim([0 50]); ylabel('Percentage (%)');
legend('Class 1','Class 4','Class 5','Class 6','Class 8')
title('ORNL')
%print(f3,'-dtiff','-r500','Monthly_Class_Pattern_ORNL');

```

A.2.4. 3-dimensional CDFs for K_D and POP- K_D at SRRL BMS, UNLV, and ORNL (Section 3.4.4)

● 3DSeasonalCDFs_SRRLBMS_UNLV_ORNL.m

```

%% 3-D Seasonal Cumulative Frequency Distribution
% This script observes 3-dimensional cumulative frequency distributions
% at different four seasons for KD and POP-KD.

clear all; close all; clc;

load Golden_2001_2011_YearlyGroup_idx_perK_proS.mat
load UNLV_2007_2011_YearlyGroup_idx_perK_proS.mat
load ORNL_2008_2011_YearlyGroup_idx_perK_proS.mat

%% Data selection
% Choose one of the below three locations
%loca = 'Golden';
%loca = 'ORNL';

```



```

loca = 'UNLV';

if strcmp(loca,'Golden')
    begYr = 2001;
elseif strcmp(loca,'UNLV')
    begYr = 2007;
elseif strcmp(loca,'ORNL')
    begYr = 2008;
else
    disp('Location error!');
end

%% Monthly seperation of data
nDysPerMn = [31 28 31 30 31 30 31 31 30 31 30 31];

endYr = 2011; % The ending year

selYr = endYr - begYr + 1; % selected years

perK_Mnly = zeros(12,31*selYr);
proS_Mnly = zeros(12,31*selYr);

for i = 1:selYr
    refYr = begYr + (i-1);
    yrPerK = eval(strcat('perK_',loca,'_',num2str(refYr)));
    yrProS = eval(strcat('proS_',loca,'_',num2str(refYr)));

    preDySum = 0;
    for j = 1:12
        perK_Mnly(j,(i-1)*nDysPerMn(j)+1:i*nDysPerMn(j))=yrPerK(preDySum+1:preDySum+nDysPerMn(j));
        proS_Mnly(j,(i-1)*nDysPerMn(j)+1:i*nDysPerMn(j))=yrProS(preDySum+1:preDySum+nDysPerMn(j));
        preDySum = preDySum + nDysPerMn(j);
    end
end

perK_01Jan = perK_Mnly(1,1:selYr*nDysPerMn(1));
perK_02Feb = perK_Mnly(2,1:selYr*nDysPerMn(2));
perK_03Mar = perK_Mnly(3,1:selYr*nDysPerMn(3));
perK_04Apr = perK_Mnly(4,1:selYr*nDysPerMn(4));
perK_05May = perK_Mnly(5,1:selYr*nDysPerMn(5));
perK_06Jun = perK_Mnly(6,1:selYr*nDysPerMn(6));
perK_07Jul = perK_Mnly(7,1:selYr*nDysPerMn(7));
perK_08Aug = perK_Mnly(8,1:selYr*nDysPerMn(8));
perK_09Sep = perK_Mnly(9,1:selYr*nDysPerMn(9));
perK_10Oct = perK_Mnly(10,1:selYr*nDysPerMn(10));
perK_11Nov = perK_Mnly(11,1:selYr*nDysPerMn(11));
perK_12Dec = perK_Mnly(12,1:selYr*nDysPerMn(12));

proS_01Jan = proS_Mnly(1,1:selYr*nDysPerMn(1));
proS_02Feb = proS_Mnly(2,1:selYr*nDysPerMn(2));
proS_03Mar = proS_Mnly(3,1:selYr*nDysPerMn(3));
proS_04Apr = proS_Mnly(4,1:selYr*nDysPerMn(4));
proS_05May = proS_Mnly(5,1:selYr*nDysPerMn(5));

```

```

proS_06Jun = proS_Mnly(6,1:selYr*nDysPerMn(6));
proS_07Jul = proS_Mnly(7,1:selYr*nDysPerMn(7));
proS_08Aug = proS_Mnly(8,1:selYr*nDysPerMn(8));
proS_09Sep = proS_Mnly(9,1:selYr*nDysPerMn(9));
proS_10Oct = proS_Mnly(10,1:selYr*nDysPerMn(10));
proS_11Nov = proS_Mnly(11,1:selYr*nDysPerMn(11));
proS_12Dec = proS_Mnly(12,1:selYr*nDysPerMn(12));

%% Choose months or seasons for comparison
% Summer
lenSum = (nDysPerMn(5) + nDysPerMn(6) + nDysPerMn(7) + nDysPerMn(8))*selYr;
perK_Sum = zeros(1,lenSum);
proS_Sum = zeros(1,lenSum);
endIdx = 0;
perK_Sum(endIdx+1:endIdx+nDysPerMn(5)*selYr) = perK_05May;
proS_Sum(endIdx+1:endIdx+nDysPerMn(5)*selYr) = proS_05May;
endIdx = nDysPerMn(5)*selYr;
perK_Sum(endIdx+1:endIdx+nDysPerMn(6)*selYr) = perK_06Jun;
proS_Sum(endIdx+1:endIdx+nDysPerMn(6)*selYr) = proS_06Jun;
endIdx = (nDysPerMn(5)+nDysPerMn(6))*selYr;
perK_Sum(endIdx+1:endIdx+nDysPerMn(7)*selYr) = perK_07Jul;
proS_Sum(endIdx+1:endIdx+nDysPerMn(7)*selYr) = proS_07Jul;
endIdx = (nDysPerMn(5)+nDysPerMn(6)+nDysPerMn(7))*selYr;
perK_Sum(endIdx+1:endIdx+nDysPerMn(8)*selYr) = perK_08Aug;
proS_Sum(endIdx+1:endIdx+nDysPerMn(8)*selYr) = proS_08Aug;

% Winter
lenWin = (nDysPerMn(1) + nDysPerMn(2) + nDysPerMn(11) + nDysPerMn(12))*selYr;
perK_Win = zeros(1,lenWin);
proS_Win = zeros(1,lenWin);
endIdx = 0;
perK_Win(endIdx+1:endIdx+nDysPerMn(1)*selYr) = perK_01Jan;
proS_Win(endIdx+1:endIdx+nDysPerMn(1)*selYr) = proS_01Jan;
endIdx = nDysPerMn(1)*selYr;
perK_Win(endIdx+1:endIdx+nDysPerMn(2)*selYr) = perK_02Feb;
proS_Win(endIdx+1:endIdx+nDysPerMn(2)*selYr) = proS_02Feb;
endIdx = (nDysPerMn(1)+nDysPerMn(2))*selYr;
perK_Win(endIdx+1:endIdx+nDysPerMn(11)*selYr) = perK_11Nov;
proS_Win(endIdx+1:endIdx+nDysPerMn(11)*selYr) = proS_11Nov;
endIdx = (nDysPerMn(1)+nDysPerMn(2)+nDysPerMn(11))*selYr;
perK_Win(endIdx+1:endIdx+nDysPerMn(12)*selYr) = perK_12Dec;
proS_Win(endIdx+1:endIdx+nDysPerMn(12)*selYr) = proS_12Dec;

% Spring + Fall
lenSnF = (nDysPerMn(3) + nDysPerMn(4) + nDysPerMn(9) + nDysPerMn(10))*selYr;
perK_SnF = zeros(1,lenSnF);
proS_SnF = zeros(1,lenSnF);
endIdx = 0;
perK_SnF(endIdx+1:endIdx+nDysPerMn(3)*selYr) = perK_03Mar;
proS_SnF(endIdx+1:endIdx+nDysPerMn(3)*selYr) = proS_03Mar;
endIdx = nDysPerMn(3)*selYr;
perK_SnF(endIdx+1:endIdx+nDysPerMn(4)*selYr) = perK_04Apr;
proS_SnF(endIdx+1:endIdx+nDysPerMn(4)*selYr) = proS_04Apr;
endIdx = (nDysPerMn(3)+nDysPerMn(4))*selYr;
perK_SnF(endIdx+1:endIdx+nDysPerMn(9)*selYr) = perK_09Sep;

```

```

proS_SnF(endIdx+1:endIdx+nDysPerMn(9)*selYr) = proS_09Sep;
endIdx = (nDysPerMn(3)+nDysPerMn(4)+nDysPerMn(9))*selYr;
perK_SnF(endIdx+1:endIdx+nDysPerMn(10)*selYr) = perK_10Oct;
proS_SnF(endIdx+1:endIdx+nDysPerMn(10)*selYr) = proS_10Oct;

%% CDF
% Summer
[f1_Sum,x1_Sum] = ecdf(perK_Sum);
[f2_Sum,x2_Sum] = ecdf(proS_Sum);

lenInterp = length(proS_Sum);

flipl_Sum = 0:1/lenInterp:1;
xlipl_Sum = interp1(f1_Sum,x1_Sum,flipl_Sum);

f2ipl_Sum = 0:1/lenInterp:1;
x2ipl_Sum = interp1(f2_Sum,x2_Sum,f2ipl_Sum);

fq_Sum = f1_Sum;

% Winter
[f1_Win,x1_Win] = ecdf(perK_Win);
[f2_Win,x2_Win] = ecdf(proS_Win);

lenInterp = length(proS_Win);

flipl_Win = 0:1/lenInterp:1;
xlipl_Win = interp1(f1_Win,x1_Win,flipl_Win);

f2ipl_Win = 0:1/lenInterp:1;
x2ipl_Win = interp1(f2_Win,x2_Win,f2ipl_Win);

fq_Win = f1_Win;

% Spring + Fall
[f1_SnF,x1_SnF] = ecdf(perK_SnF);
[f2_SnF,x2_SnF] = ecdf(proS_SnF);

lenInterp = length(proS_SnF);

flipl_SnF = 0:1/lenInterp:1;
xlipl_SnF = interp1(f1_SnF,x1_SnF,flipl_SnF);

f2ipl_SnF = 0:1/lenInterp:1;
x2ipl_SnF = interp1(f2_SnF,x2_SnF,f2ipl_SnF);

fq_SnF = f1_SnF;

%% Plot
liWidth = 2;
fracStr=char('Cumulative Frequency (F)');

% 3-D CDF

```

```

f1 = figure(1);
plot3(xl1pl_Sum,x2ipl_Sum,fq_Sum,'r-','LineWidth',liWidth)
grid on
hold on
plot3(xl1pl_Win,x2ipl_Win,fq_Win,'b-','LineWidth',liWidth)
plot3(xl1pl_SnF,x2ipl_SnF,fq_SnF,'k-','LineWidth',liWidth)
hold off
xlabel('K_D');
ylabel('POP-K_D');
zlabel(fracStr);
xlim([0 0.9]);ylim([0.5 1])
legend('Summer','Winter','Spring&Fall','Location','East')
title('(a) 3D');
picName1 = strcat('Seasonal_3DCDF_',loca);
%print(f1,'-dtiff','-r500',picName1);

% 2-D projection of KD and POP-KD
f2 = figure(2);
subplot(121);
plot(fq_Sum,xl1pl_Sum,'r-','LineWidth',liWidth)
grid on
hold on
plot(fq_Win,xl1pl_Win,'b-','LineWidth',liWidth)
plot(fq_SnF,xl1pl_SnF,'k-','LineWidth',liWidth)
hold off
xlabel(fracStr);
ylabel('K_D');
xlim([0 1]);ylim([0 1])
legend('Summer','Winter','Spring&Fall','Location','SouthEast')
title('(b) 2D (K_D)')

subplot(122);
plot(fq_Sum,x2ipl_Sum,'r-','LineWidth',liWidth)
grid on
hold on
plot(fq_Win,x2ipl_Win,'b-','LineWidth',liWidth)
plot(fq_SnF,x2ipl_SnF,'k-','LineWidth',liWidth)
hold off
xlabel(fracStr);
ylabel('POP-K_D');
xlim([0 1]);ylim([0.5 1])
legend('Summer','Winter','Spring&Fall','Location','SouthEast')
title('(c) 2D (POP-K_D)')
picName23 = strcat('Seasonal_twoCDFs_',loca);
%print(f2,'-dtiff','-r500',picName23);

% KD and POP-KD
f3 = figure(3);
plot(xl1pl_Sum,x2ipl_Sum,'r-','LineWidth',liWidth)
grid on
hold on
plot(xl1pl_Win,x2ipl_Win,'b-','LineWidth',liWidth)
plot(xl1pl_SnF,x2ipl_SnF,'k-','LineWidth',liWidth)
hold off
xlabel('K_D')
ylabel('POP-K_D');

```

```

xlim([0 0.9]);ylim([0.5 1])
legend('Summer','Winter','Spring&Fall','Location','SouthEast')
title('(d) 2D (K_D & POP-K_D)')
picName4 = strcat('Seasonal_KDnPOP-KD_',loca);
%print(f3,'-dtiff','-r500',picName4);

```

A.2.5. Generation of synthetic sequences (Section 3.5)

● SequenceGeneration_KD_SRRLBMS_UNLV_ORNL.m

```

%% Generation of Synthetic Sequences for KD
% This script generates an yearly synthetic sequence for KD based on the
% generated library of transition matrices for KD at SRRL BMS, UNLV, and
% ORNL according to Aguiar et al. (1998).

clear all; close all; clc;

load Golden_2001_2011_YearlyGroup_idx_perK_proS.mat
load UNLV_2007_2011_YearlyGroup_idx_perK_proS.mat
load ORNL_2008_2011_YearlyGroup_idx_perK_proS.mat
load AguiarMTM_All_Locations_Monthly_perK_proS_classes.mat

%% Selecting location
% % Golden, 2011
% location = 'Golden';
% yr = 2011;
% refYr = yr - 2001 + 1;
% begMn = (refYr - 1)*12 + 1;
% endMn = begMn + 12 - 1;
% perK12Mn = mnAvePerKAll(begMn:endMn);

% % UNLV, 2011
% location = 'UNLV';
% yr = 2011;
% refYr = yr - 2007 + 12;
% begMn = (refYr - 1)*12 + 1;
% endMn = begMn + 12 - 1;
% perK12Mn = mnAvePerKAll(begMn:endMn);

% ORNL, 2011
location = 'ORNL';
yr = 2011;
refYr = yr - 2008 + 17;
begMn = (refYr - 1)*12 + 1;
endMn = begMn + 12 - 1;
perK12Mn = mnAvePerKAll(begMn:endMn);

%% Classification
cls12Mn = zeros(1,12);
for i = 1:12
    cls12Mn(i) = classAguiarMTM(perK12Mn(i), 'perK');

```

```

end
%% Sequence generation
nDysPerMn = [31 28 31 30 31 30 31 31 30 31 30 31];

perKMn = perK12Mn;
clsMn = cls12Mn;

genSeqYr = zeros(1,365);

mnDySumIdx = 0;
for mn = 1:12
    if mn == 1
        iniRef = perKMn(12);
    else
        iniRef = perKMn(mn-1);
    end

    mtm = eval(strcat('MTMperK.class',num2str(clsMn(mn))));

    genSeqMn = zeros(1,nDysPerMn(mn));
    for dy = 1:nDysPerMn(mn)
        if dy == 1
            prevDyVal = iniRef;
        else
            prevDyVal = genSeqMn(dy-1);
        end

        ranNum = rand(1);
        rIdx = classQuanAugiar(prevDyVal,clsMn(mn),'perK');

        rSum = 0;
        for col = 1:10
            rSum = rSum + mtm(rIdx,col);
            if rSum > ranNum
                if col == 1
                    proLow = 0;
                else
                    proLow = sum(mtm(rIdx,1:col-1));
                end
                proHig = sum(mtm(rIdx,1:col));
                cIdx = col;
                break;
            end
        end
        [valLow,valHig] = findBoundQuanAugiar(cIdx,clsMn(mn),'perK');

        genSeqMn(dy) = (valHig-valLow)/(proHig-proLow)*(ranNum-proLow)+valLow;
    end
    genSeqYr(mnDySumIdx+1:mnDySumIdx+nDysPerMn(mn)) = genSeqMn;

    mnDySumIdx = mnDySumIdx + nDysPerMn(mn);
end

%% KSTest

```

```

obsSeqYr = eval(strcat('perK_',location,'_',num2str(yr)));
[h,p,k] = kstest2(obsSeqYr,genSeqYr)

%% Plot
figure(1);
subplot(211);plot(1:365,obsSeqYr)
xlabel('Days');ylabel('K_D')
title(strcat('Observed sequence,',location,'(',num2str(yr),')'))
xlim([1 365]);ylim([0 1])

subplot(212);plot(1:365,genSeqYr,'r-')
xlabel('Days');ylabel('K_D')
title(strcat('Generated sequence,',location,'(',num2str(yr),')'))
xlim([1 365]);ylim([0 1])

```

● SequenceGeneration_POPKD_SRRLBMS_UNLV_ORNL.m

```

%% Generation of Synthetic Sequences for POP-KD
% This script generates an yearly synthetic sequence for POP-KD based on
% the generated library of transition matrices for POP-KD at SRRL BMS,
% UNLV, and ORNL according to Aguiar et al. (1998).

clear all; close all; clc;

load Golden_2001_2011_YearlyGroup_idx_perK_proS.mat
load UNLV_2007_2011_YearlyGroup_idx_perK_proS.mat
load ORNL_2008_2011_YearlyGroup_idx_perK_proS.mat
load AguiarMTM_All_Locations_Monthly_perK_proS_classes.mat

%% Selecting location
% % Golden, 2011
% location = 'Golden';
% yr = 2011;
% refYr = yr - 2001 + 1;
% begMn = (refYr - 1)*12 + 1;
% endMn = begMn + 12 - 1;
% proS12Mn = mnAveProSAll(begMn:endMn);

% % UNLV, 2011
% location = 'UNLV';
% yr = 2011;
% refYr = yr - 2007 + 12;
% begMn = (refYr - 1)*12 + 1;
% endMn = begMn + 12 - 1;
% proS12Mn = mnAveProSAll(begMn:endMn);

% ORNL, 2011
location = 'ORNL';
yr = 2011;
refYr = yr - 2008 + 17;
begMn = (refYr - 1)*12 + 1;
endMn = begMn + 12 - 1;
proS12Mn = mnAveProSAll(begMn:endMn);

```

```

%% Classification
cls12Mn = zeros(1,12);
for i = 1:12
    cls12Mn(i) = classAguiarMTM(proS12Mn(i),'proS');
end
%% Sequence generation
nDysPerMn = [31 28 31 30 31 30 31 31 30 31 30 31];

proSMn = proS12Mn;
clsMn = cls12Mn;

genSeqYr = zeros(1,365);

mnDySumIdx = 0;
for mn = 1:12
    if mn == 1
        iniRef = proSMn(12);
    else
        iniRef = proSMn(mn-1);
    end

    mtm = eval(strcat('MTMproS.class',num2str(clsMn(mn))));

    genSeqMn = zeros(1,nDysPerMn(mn));
    for dy = 1:nDysPerMn(mn)
        if dy == 1
            prevDyVal = iniRef;
        else
            prevDyVal = genSeqMn(dy-1);
        end

        ranNum = rand(1);
        rIdx = classQuanAugiar(prevDyVal,clsMn(mn),'proS');

        rSum = 0;
        for col = 1:10
            rSum = rSum + mtm(rIdx,col);
            if rSum > ranNum
                if col == 1
                    proLow = 0;
                else
                    proLow = sum(mtm(rIdx,1:col-1));
                end
                proHig = sum(mtm(rIdx,1:col));
                cIdx = col;
                break;
            end
        end
        [valLow,valHig] = findBoundQuanAugiar(cIdx,clsMn(mn),'proS');

        genSeqMn(dy) = (valHig-valLow)/(proHig-proLow)*(ranNum-proLow)+valLow;
    end
    genSeqYr(mnDySumIdx+1:mnDySumIdx+nDysPerMn(mn)) = genSeqMn;

```



```

        mnDySumIdx = mnDySumIdx + nDysPerMn(mn);
end

%% KSTest
obsSeqYr = eval(strcat('pros_',location,'_',num2str(yr)));
[h,p,k] = kstest2(obsSeqYr,genSeqYr)

%% Plot
figure(1);
subplot(211);plot(1:365,obsSeqYr)
xlabel('Days');ylabel('POP_D')
title(strcat('Observed sequence,',location,'(',num2str(yr),')'))
xlim([1 365])

subplot(212);plot(1:365,genSeqYr,'r-')
xlabel('Days');ylabel('POP_D')
title(strcat('Generated sequence,',location,'(',num2str(yr),')'))
xlim([1 365])

```

● classAguiarMTM.m

```

%% Function to Classfy Monthly Averages (Augiar MTM method)
% This function classifies the provided monthly average value into ten
% classes following Augiar's MTM approach.

function output = classAguiarMTM(input, dataStr)
%% Input
% input: monthly average
% dataStr: string for indicating either 'perK' or 'proS'

%% Output
% output: classified number from one to ten

%% Boundary and gap setup

lowBd_perK = 0.40;
gap_perK = 0.05;

lowBd_proS = 0.70;
gap_proS = 0.025;

%% Classification
% perK
if strcmp(dataStr,'perK')
    if input <= lowBd_perK
        output = 1;
    elseif input > lowBd_perK + gap_perK*0 && input <= lowBd_perK +
gap_perK*1
        output = 2;
    elseif input > lowBd_perK + gap_perK*1 && input <= lowBd_perK +
gap_perK*2
        output = 3;

```

```

        elseif input > lowBd_perK + gap_perK*2 && input <= lowBd_perK +
gap_perK*3
            output = 4;
        elseif input > lowBd_perK + gap_perK*3 && input <= lowBd_perK +
gap_perK*4
            output = 5;
        elseif input > lowBd_perK + gap_perK*4 && input <= lowBd_perK +
gap_perK*5
            output = 6;
        elseif input > lowBd_perK + gap_perK*5 && input <= lowBd_perK +
gap_perK*6
            output = 7;
        elseif input > lowBd_perK + gap_perK*6 && input <= lowBd_perK +
gap_perK*7
            output = 8;
        elseif input > lowBd_perK + gap_perK*7 && input <= lowBd_perK +
gap_perK*8
            output = 9;
        else
            output = 10;
        end

% proS
elseif strcmp(dataStr,'proS')
    if input <= lowBd_proS
        output = 1;
    elseif input > lowBd_proS + gap_proS*0 && input <= lowBd_proS +
gap_proS*1
        output = 2;
    elseif input > lowBd_proS + gap_proS*1 && input <= lowBd_proS +
gap_proS*2
        output = 3;
    elseif input > lowBd_proS + gap_proS*2 && input <= lowBd_proS +
gap_proS*3
        output = 4;
    elseif input > lowBd_proS + gap_proS*3 && input <= lowBd_proS +
gap_proS*4
        output = 5;
    elseif input > lowBd_proS + gap_proS*4 && input <= lowBd_proS +
gap_proS*5
        output = 6;
    elseif input > lowBd_proS + gap_proS*5 && input <= lowBd_proS +
gap_proS*6
        output = 7;
    elseif input > lowBd_proS + gap_proS*6 && input <= lowBd_proS +
gap_proS*7
        output = 8;
    elseif input > lowBd_proS + gap_proS*7 && input <= lowBd_proS +
gap_proS*8
        output = 9;
    else
        output = 10;
    end
else
    disp('The input data is not specified either perK or proS!');
end

```

● classQuanAugiar.m

```
%% Function to Classify Quantization Level Based on Augiar's Class
% This function classifies the quantization level using provided min & max
% values of each class that classified by the Augiar's MTM approach.

function qClass = classQuanAugiar(input, class, dataStr)
%% Input
% input: daily values
% class: classified value based on Augiar's MTM approach for each month
% dataStr: string for indicating either 'perK' or 'proS'

%% Output
% qClass: quantized classification result

%% Read monthly min and max value for each class
load AugiarMTM_All_Locations_Monthly_perK_proS_classes.mat

if strcmp(dataStr, 'perK')
    maxCls = clsMaxPerKAll(class);
    minCls = clsMinPerKAll(class);
elseif strcmp(dataStr, 'proS')
    maxCls = clsMaxProSAll(class);
    minCls = clsMinProSAll(class);
else
    disp('The input data is not specified either perK or proS!');
end

gap = (maxCls - minCls) / 10;
%% Classification
if input <= minCls + gap*1
    qClass = 1;
elseif input > minCls + gap*1 && input <= minCls + gap*2
    qClass = 2;
elseif input > minCls + gap*2 && input <= minCls + gap*3
    qClass = 3;
elseif input > minCls + gap*3 && input <= minCls + gap*4
    qClass = 4;
elseif input > minCls + gap*4 && input <= minCls + gap*5
    qClass = 5;
elseif input > minCls + gap*5 && input <= minCls + gap*6
    qClass = 6;
elseif input > minCls + gap*6 && input <= minCls + gap*7
    qClass = 7;
elseif input > minCls + gap*7 && input <= minCls + gap*8
    qClass = 8;
elseif input > minCls + gap*8 && input <= minCls + gap*9
    qClass = 9;
else
    qClass = 10;
end
```

● findBoundQuanAugiar.m

```

%% Function to find Boundaries Based on Augiar's Class
% This function searches the high and low boundaries for the quantized
% classes.

function [lowBound,uppBound] = findBoundQuanAugiar(colIdx, class, dataStr)
%% Input
% input: daily values
% class: classified value based on Augiar's MTM approach for each month
% dataStr: string for indicating either 'perK' or 'proS'

%% Output
% lowBound: lower boundaries for the provided column index
% uppBound: upper boundaries for the provided column index

%% Read monthly min and max value for each class
load AugiarMTM_All_Locations_Monthly_perK_proS_classes.mat

if strcmp(dataStr,'perK')
    maxCls = clsMaxPerKAll(class);
    minCls = clsMinPerKAll(class);
elseif strcmp(dataStr,'proS')
    maxCls = clsMaxProSAll(class);
    minCls = clsMinProSAll(class);
else
    disp('The input data is not specified either perK or proS!');
end

gap = (maxCls - minCls) / 10;

%% High & low boundaries for the selected quantized class
if colIdx == 1
    lowBound = minCls;
    uppBound = lowBound + gap;

elseif colIdx == 10
    lowBound = minCls + gap*9;
    uppBound = lowBound + gap;
else
    uppBound = minCls + gap*colIdx;
    lowBound = minCls + gap*(colIdx-1);
end

```

A.3. M-files for Chapter 4

A.3.1. Bias between $TKC_{D,NWS}$ and TKC_D (Section 4.3)

● BiasTKCDforTKCDobs.m

```
%% Bias Observation between TKCD,NWS and TKCD
% This script observes the bias between daily forecast total sky cover
% (TKCD,NWS) and daily observed total sky cover (TKCD) at SRRL BMS from
% 2009 to 2011.

clear all; close all; clc;

load Golden_2009_2012_YearlyGroup_ClassGroup_TKCDfor_TKCDobs.mat

%% Bias between forecasted and observed TKCD
biasForObsTCCD = TCCDfor_0911 - TCCDobs_0911;
biasForObsTCCD_hig = TCCDfor_0911_cls147 - TCCDobs_0911_cls147;
biasForObsTCCD_mid = TCCDfor_0911_cls258 - TCCDobs_0911_cls258;
biasForObsTCCD_low = TCCDfor_0911_cls369 - TCCDobs_0911_cls369;

%% Plotting distributions
nBins2 = 15;
f1 = figure(1);
subplot(221);hist(biasForObsTCCD,nBins2);xlim([-0.9 0.9])
ylabel('Density'); title('(a) Overall')
subplot(222);hist(biasForObsTCCD_hig,nBins2);xlim([-0.9 0.9]);
ylabel('Density'); title('(b) High K_D')
subplot(223);hist(biasForObsTCCD_mid,nBins2);xlim([-0.9 0.9])
ylabel('Density'); title('(c) Medium K_D')
subplot(224);hist(biasForObsTCCD_low,10);xlim([-0.9 0.9]);
ylabel('Density'); title('(d) Low K_D')
%print(f1,'-dtiff','-r500','HistogramBiasForObsTCCD');
```

A.3.2. Curve fitting between TKC_D and K_D (Section 4.4)

● CurveFittingTKCDnKDBoxCox.m

```
%% Curve Fitting Using Box-Cox Transform
% This script fits the conversion curve from daily observed total sky
% cover (TKCD) to daily solar irradiance (KD) at SRRL BMS for 2009 to
% 2011.
% The Box-Cox transform is used for the curve fitting.

clear all; close all; clc;

load Golden_2009_2012_YearlyGroup_ClassGroup_TKCDfor_TKCDobs.mat
```

```

%% Box-Cox transform

ydata = KD_0911;
xdata = TCCDobs_0911;

[transdat,lambda] = boxcox(ydata); %Box-Cox transform

xx = 0:0.01:1;
yy_tran = -0.2858*xx-0.1979;
yy_tran_up = -0.2858*xx-0.1979+1.96*0.0565; %upper boundary of 95% CI
yy_tran_lo = -0.2858*xx-0.1979-1.96*0.0565; %lower boundary of 95% CI
for dataIdx = 1:3
    if dataIdx == 1
        data = yy_tran;
    elseif dataIdx == 2
        data = yy_tran_up;
    else
        data = yy_tran_lo;
    end

    yy_invT = zeros(size(xx));
    for i = 1:length(xx)
        if lambda*data(i) > -1
            yy_invT(i) = (lambda*data(i)+1).^(1/lambda);
        else
            yy_invT(i) = 0;
        end
    end

    if dataIdx == 1
        yy = yy_invT;
    elseif dataIdx == 2
        yy_up = yy_invT;
    else
        yy_lo = yy_invT;
    end
end

%% Plotting
mkSize = 15;
f2 = figure(2);
plot(xdata,transdat,'k.','MarkerSize',mkSize)
hold on
plot(xx,yy_tran,'r-','LineWidth',2)
hold off
legend('Data points','Fitted line')
xlabel('TKC_D'); ylabel('K_D_t_r_a_n_s')
%print(f2,'-dtiff','-r500','LineFittingTransformed');

f3 = figure(3);
plot(xdata,ydata,'k.','MarkerSize',mkSize)
hold on
plot(xx,yy,'r-','LineWidth',2)
hold off

```

```

legend('Data points','Fitted curve')
xlabel('TKC_D'); ylabel('K_D')
%print(f3,'-dtiff','-r500','LineFittingInverseTransformed');

```

A.3.3. Description of the proposed Methods 1 – 5 (Section 4.5)

● CIBoundaries_Methods12345.m

```

%% Conversion Curves with 95% CIs for Methods 1 - 5
% This script plots figures that the proposed conversion curve with its
% 95% CI boundaries for Methods 1, 2, 3, 4, and 5.
% This script also plots the figure that describes the gap generated by
% adjustment in Stage 2A.

clear all; close all; clc;

a = -0.2825; b = -0.1979;
lambda = 2.0906;

xx = 0.001:0.001:1-0.001;
yy = (lambda*(a*xx+b)+1).^(1/lambda);
yy_up = (lambda*(a*xx+b+1.96*0.0565)+1).^(1/lambda);
yy_lo = (lambda*(a*xx+b-1.96*0.0565)+1).^(1/lambda);

obsPnt = 300; %0.3

%% Figure for conversion curve with 95% CI boundaries for Method 1234
f1 = figure(1);
plot(xx,yy_up,'r--','LineWidth',1)
hold on
plot(xx,yy,'r-','LineWidth',2)
plot(xx(1:601),yy_lo(1:601),'r--','LineWidth',1)
% estimation
plot(xx(obsPnt),0:0.01:yy(obsPnt),'k.','LineWidth',.5)
plot(0:0.01:xx(obsPnt),yy(obsPnt),'k.','LineWidth',.5)
text(0.01,yy(obsPnt)+0.03,'K_D,_f_o_r','FontSize',10)
% upper
plot(xx(obsPnt),0:0.01:yy_up(obsPnt),'k.','LineWidth',.5)
plot(0:0.01:xx(obsPnt),yy_up(obsPnt),'k.','LineWidth',.5)
text(0.01,yy_up(obsPnt)+0.03,'Upper boundary of K_D,_f_o_r','FontSize',10)
% lower
plot(xx(obsPnt),0:0.01:yy_lo(obsPnt),'k.','LineWidth',.5)
plot(0:0.01:xx(obsPnt),yy_lo(obsPnt),'k.','LineWidth',.5)
text(0.01,yy_lo(obsPnt)+0.03,'Lower boundary of K_D,_f_o_r','FontSize',10)
hold off
legend('95% CI (upper)','Conversion curve',...
      '95% CI (lower)')
xlabel('TKC_D,_f_o_r'); ylabel('K_D,_f_o_r')
ylim([0.001 1])
title('(a) Method 1, 2, 3, and 4')

```

```

%print(f1,'-dtiff','-r500','UncertaintyBoundaryMethod1234');

%% Figure for conversion curve with 95% CI boundaries for Method 5
yy_up(1:411) = yy_up(1:411) + 0.0559;
yy(1:411) = yy(1:411) + 0.0559;
yy_lo(1:411) = yy_lo(1:411) + 0.0559;

f2 = figure(2);
plot(xx,yy_up,'r--','LineWidth',1)
hold on
plot(xx,yy,'r-','LineWidth',2)
plot(xx(1:601),yy_lo(1:601),'r--','LineWidth',1)
% estimation
plot(xx(obsPnt),0:0.01:yy(obsPnt),'k.','LineWidth',.5)
plot(0:0.01:xx(obsPnt),yy(obsPnt),'k.','LineWidth',.5)
text(0.01,yy(obsPnt)+0.03,'K_D,_f_o_r','FontSize',10)
text(0.42,0.6264,'\uparrow 0.0559 UP','FontSize',10)
% upper
plot(xx(obsPnt),0:0.01:yy_up(obsPnt),'k.','LineWidth',.5)
plot(0:0.01:xx(obsPnt),yy_up(obsPnt),'k.','LineWidth',.5)
text(0.01,yy_up(obsPnt)+0.03,'Upper boundary of K_D,_f_o_r','FontSize',10)
text(0.42,(.8234+.7671)/2,'\uparrow 0.0559 UP','FontSize',10)
% lower
plot(xx(obsPnt),0:0.01:yy_lo(obsPnt),'k.','LineWidth',.5)
plot(0:0.01:xx(obsPnt),yy_lo(obsPnt),'k.','LineWidth',.5)
text(0.01,yy_lo(obsPnt)+0.03,'Lower boundary of K_D,_f_o_r','FontSize',10)
text(0.42,(.4069+.3501)/2,'\uparrow 0.0559 UP','FontSize',10)
hold off
legend('95% CI (upper)','Conversion curve',...
       '95% CI (lower)')
xlabel('TKC_D,_f_o_r'); ylabel('K_D,_f_o_r')
ylim([0.001 1])
title('(c) Method 5')
%print(f2,'-dtiff','-r500','UncertaintyBoundaryMethod5');

%% Figure for describing the gap generated by adjustment in Stage 2A
f3 = figure(3);
yy(1:411) = yy(1:411) - 0.0559;
plot(xx(1:10:411),yy(1:10:411),'r--','LineWidth',1)
hold on
plot(xx(418:length(xx)),yy(418:length(xx)),'r-','LineWidth',2)
plot(xx(1:411),yy(1:411)+0.0559,'r-','LineWidth',2)
plot(xx(411),yy(411):0.01:yy(411)+0.0559,'r.','LineWidth',.5)

plot(xx(411),0:0.01:yy(411),'k.','LineWidth',.5)
text(0.42,0.6264,'\uparrow 0.0559 UP','FontSize',10)
text(0.32,0.3,'Gap','FontSize',15)
text(0.32,0.15,'\leftarrow \mu_a_d_j','FontSize',10)

plot(xx(296),0:0.01:yy(296),'k.','LineWidth',.5)
text(xx(296)+0.01,0.02,'0.2958','FontSize',10)
text(xx(411)+0.01,0.02,'0.4106','FontSize',10)
text(0.005,0.7746+0.008,'0.7746','FontSize',10)
text(0.005,0.8305+0.008,'0.8305','FontSize',10)
hold off
Legend('Original curve','Shifted curve')

```



```

xlabel('TKC_D'); ylabel('K_D,_f_o_r')
title('(b) Gap by adjustment')

%print(f3,'-dtiff','-r500','DifferentConversionCurves');

```

A.3.4. Tests for the proposed Methods 1 – 5 (Section 4.6)

● EstimationTest_Method1_2012.m

```

%% Estimation Test of Method 1 Using the 2012 Data
% This script tests the estimation performance of the proposed Method 1
% using the 2012 data. The RMSE, MBE, KSI, OVER, and hit rate are observed
% for the performance.

clear all; close all; clc;

load Golden_2009_2012_YearlyGroup_ClassGroup_TKCDfor_TKCDobs.mat

%% Data selection
yr = 2012;
KD_selYr = eval(strcat('KD_',num2str(yr)));
refKD = KD_selYr;

TKCDfor_selYr = eval(strcat('TKCDfor_',num2str(yr)));
TKCDfor = TKCDfor_selYr;

numDysYr = length(KD_selYr(:,1));

%% Estimating KD
% Relationship between KD and TKCDobs:
% KDhat_trans = -0.2825*xdata - 0.1979;
% KDhat = (lambda*KDhat_trans+1)^(1/lambda);

lambda = 2.0906;

TKCDfor = TKCDfor;
u_TKCDfor = 0.2;
u_a = 0.002908; u_b = 0.001327;
a = -0.2825; b = -0.1979;
sigma = 0.0565;

estKD = zeros(numDysYr,1);
estKD_up = zeros(numDysYr,1);
estKD_lo = zeros(numDysYr,1);

for i = 1:numDysYr
    % estimation
    if (-0.2825*TKCDfor(i)-0.1979)*lambda > -1
        estKD(i) = (lambda*(-0.2825*TKCDfor(i)-0.1979)+1)^(1/lambda);
    end
end

```

```

else
    estKD(i) = 0;
end

% upper
if (-0.2825*TKCDfor(i)-0.1979+1.96*sigma)*lambda > -1
    estKD_up(i) = (lambda*(-0.2825*TKCDfor(i)-
0.1979+1.96*sigma)+1)^(1/lambda);
else
    estKD_up(i) = 0;
end

% lower
if (-0.2825*TKCDfor(i)-0.1979-1.96*sigma)*lambda > -1
    estKD_lo(i) = (lambda*(-0.2825*TKCDfor(i)-0.1979-
1.96*sigma)+1)^(1/lambda);
else
    estKD_lo(i) = 0;
end
end

%% Classification by high, mid, and low KD level
numDys_hig = 0; numDys_mid = 0; numDys_low = 0;
idxDys_hig = zeros(numDysYr,1);
idxDys_mid = zeros(numDysYr,1);
idxDys_low = zeros(numDysYr,1);

for i = 1:numDysYr
    if refKD(i) >= 0.6
        numDys_hig = numDys_hig + 1;
        idxDys_hig(numDys_hig) = i;
    elseif refKD(i) < 0.3
        numDys_low = numDys_low + 1;
        idxDys_low(numDys_low) = i;
    else
        numDys_mid = numDys_mid + 1;
        idxDys_mid(numDys_mid) = i;
    end
end

idxDys_hig = idxDys_hig(1:numDys_hig);
idxDys_mid = idxDys_mid(1:numDys_mid);
idxDys_low = idxDys_low(1:numDys_low);

refKD_hig = refKD; refKD_hig = refKD_hig(idxDys_hig(:,1));
refKD_mid = refKD; refKD_mid = refKD_mid(idxDys_mid(:,1));
refKD_low = refKD; refKD_low = refKD_low(idxDys_low(:,1));

estKD_hig = estKD; estKD_hig = estKD_hig(idxDys_hig(:,1));
estKD_mid = estKD; estKD_mid = estKD_mid(idxDys_mid(:,1));
estKD_low = estKD; estKD_low = estKD_low(idxDys_low(:,1));

%% Result analysis for KD
idxInOut = zeros(numDysYr,3);
idxInOut(idxDys_hig(:,1),1) = 1;

```

```

idxInOut(idxDys_mid(:,1),1) = 2;
idxInOut(idxDys_low(:,1),1) = 3;

for i = 1:numDysYr
    if refKD(i)<=estKD_up(i) && refKD(i)>=estKD_lo(i)
        idxInOut(i,2) = 1;
    elseif refKD(i)>estKD_up(i)
        idxInOut(i,2) = 0;
        idxInOut(i,3) = refKD(i) - estKD_up(i);
    else
        idxInOut(i,2) = 0;
        idxInOut(i,3) = refKD(i) - estKD_lo(i);
    end
end

numIn = length(idxInOut(idxInOut(:,2)==1,2));

idxInOut_hig = idxInOut(idxInOut(:,1)==1,:);
numIn_hig = length(idxInOut_hig(idxInOut_hig(:,2)==1,2));

idxInOut_mid = idxInOut(idxInOut(:,1)==2,:);
numIn_mid = length(idxInOut_mid(idxInOut_mid(:,2)==1,2));

idxInOut_low = idxInOut(idxInOut(:,1)==3,:);
numIn_low = length(idxInOut_low(idxInOut_low(:,2)==1,2));

if sum(idxInOut(idxInOut(:,2)==0,3)) == 0
    MAOutDist = 0;
else
    MAOutDist = mean(abs(idxInOut(idxInOut(:,2)==0,3)));
    MBOutDist = mean(idxInOut(idxInOut(:,2)==0,3));
end

if sum(idxInOut_hig(idxInOut_hig(:,2)==0,3)) == 0
    MAOutDist_hig = 0;
    MBOutDist_hig = 0;
else
    MAOutDist_hig = mean(abs(idxInOut_hig(idxInOut_hig(:,2)==0,3)));
    MBOutDist_hig = mean(idxInOut_hig(idxInOut_hig(:,2)==0,3));
end

if sum(idxInOut_mid(idxInOut_mid(:,2)==0,3)) == 0
    MAOutDist_mid = 0;
    MBOutDist_mid = 0;
else
    MAOutDist_mid = mean(abs(idxInOut_mid(idxInOut_mid(:,2)==0,3)));
    MBOutDist_mid = mean(idxInOut_mid(idxInOut_mid(:,2)==0,3));
end

if sum(idxInOut_low(idxInOut_low(:,2)==0,3)) == 0
    MAOutDist_low = 0;
    MBOutDist_low = 0;
else
    MAOutDist_low = mean(abs(idxInOut_low(idxInOut_low(:,2)==0,3)));
    MBOutDist_low = mean(idxInOut_low(idxInOut_low(:,2)==0,3));
end

```

```

end

perIn = numIn/numDysYr * 100;
perIn_hig = numIn_hig/numDys_hig * 100;
perIn_mid = numIn_mid/numDys_mid * 100;
perIn_low = numIn_low/numDys_low * 100;

%% RMSE and MBE for KD
RMSE_KD = sqrt(mean((refKD - estKD).^2));
MBE_KD = mean(estKD - refKD);
MAE_KD = mean(abs(refKD - estKD));

RMSE_KD_hig = sqrt(mean((refKD_hig - estKD_hig).^2));
MBE_KD_hig = mean(refKD_hig - estKD_hig);
MAE_KD_hig = mean(abs(refKD_hig - estKD_hig));

RMSE_KD_mid = sqrt(mean((refKD_mid - estKD_mid).^2));
MBE_KD_mid = mean(refKD_mid - estKD_mid);
MAE_KD_mid = mean(abs(refKD_mid - estKD_mid));

RMSE_KD_low = sqrt(mean((refKD_low - estKD_low).^2));
MBE_KD_low = mean(refKD_low - estKD_low);
MAE_KD_low = mean(abs(refKD_low - estKD_low));

rRMSE_KD = sqrt(mean((refKD - estKD).^2)/mean(refKD.^2));
rRMSE_KD_hig = sqrt(mean((refKD_hig - estKD_hig).^2)/mean(refKD_hig.^2));
rRMSE_KD_mid = sqrt(mean((refKD_mid - estKD_mid).^2)/mean(refKD_mid.^2));
rRMSE_KD_low = sqrt(mean((refKD_low - estKD_low).^2)/mean(refKD_low.^2));

rMBE_KD = mean(estKD-refKD)/mean(refKD);
rMBE_KD_hig = mean(refKD_hig - estKD_hig)/mean(refKD_hig);
rMBE_KD_mid = mean(refKD_mid - estKD_mid)/mean(refKD_mid);
rMBE_KD_low = mean(refKD_low - estKD_low)/mean(refKD_low);

%% RMSE and MBE for TKCD
TCCDobs_selYr = eval(strcat('TCCDobs_', num2str(yr)));
refTKCD = TCCDobs_selYr;
estTKCD = TKCDfor;

refTKCD_hig = refTKCD; refTKCD_hig = refTKCD_hig(idxDys_hig(:,1));
refTKCD_mid = refTKCD; refTKCD_mid = refTKCD_mid(idxDys_mid(:,1));
refTKCD_low = refTKCD; refTKCD_low = refTKCD_low(idxDys_low(:,1));

estTKCD_hig = estTKCD; estTKCD_hig = estTKCD_hig(idxDys_hig(:,1));
estTKCD_mid = estTKCD; estTKCD_mid = estTKCD_mid(idxDys_mid(:,1));
estTKCD_low = estTKCD; estTKCD_low = estTKCD_low(idxDys_low(:,1));

RMSE_TKCD = sqrt(mean((refTKCD - estTKCD).^2));
RMSE_TKCD_hig = sqrt(mean((refTKCD_hig - estTKCD_hig).^2));
RMSE_TKCD_mid = sqrt(mean((refTKCD_mid - estTKCD_mid).^2));
RMSE_TKCD_low = sqrt(mean((refTKCD_low - estTKCD_low).^2));

rRMSE_TKCD = sqrt(mean((refTKCD - estTKCD).^2)/mean(refTKCD.^2));

```

```

rRMSE_TKCD_hig = sqrt(mean((refTKCD_hig -
estTKCD_hig).^2)/mean(refTKCD_hig.^2));
rRMSE_TKCD_mid = sqrt(mean((refTKCD_mid -
estTKCD_mid).^2)/mean(refTKCD_mid.^2));
rRMSE_TKCD_low = sqrt(mean((refTKCD_low -
estTKCD_low).^2)/mean(refTKCD_low.^2));

MBE_TKCD = mean(estTKCD - refTKCD);
MBE_TKCD_hig = mean(refTKCD_hig - estTKCD_hig);
MBE_TKCD_mid = mean(refTKCD_mid - estTKCD_mid);
MBE_TKCD_low = mean(refTKCD_low - estTKCD_low);

rMBE_TKCD = mean(estTKCD-refTKCD)/mean(refTKCD);
rMBE_TKCD_hig = mean(refTKCD_hig - estTKCD_hig)/mean(refTKCD_hig);
rMBE_TKCD_mid = mean(refTKCD_mid - estTKCD_mid)/mean(refTKCD_mid);
rMBE_TKCD_low = mean(refTKCD_low - estTKCD_low)/mean(refTKCD_low);

%% KSI and OVER calculation
%%% Interpolation of 100 points
data_set1 = KD_2012;
data_set2 = estKD;

[f_set1,x_set1] = ecdf(data_set1);
[f_set2,x_set2] = ecdf(data_set2);

x_min_set1 = min(x_set1);
x_min_set2 = min(x_set2);

x_max_set1 = max(x_set1);
x_max_set2 = max(x_set2);

if x_min_set1 < x_min_set2
    x_min = x_min_set1;
else
    x_min = x_min_set2;
end
if x_max_set1 > x_max_set2
    x_max = x_max_set1;
else
    x_max = x_max_set2;
end

x_sam = (x_max - x_min) / 100;

xi = x_min:x_sam:x_max-x_sam;

f_inpl_set1 = zeros(100,1);
f_inpl_set2 = zeros(100,1);

for i = 1:100
    x_ref = xi(i);
    % set1
    for j = 2:length(x_set1)
        if x_ref < x_set1(j-1)
            f_inpl_set1(i) = 0;

```

```

        break;
    elseif x_ref >= x_set1(j-1) && x_ref < x_set1(j)
        f_inpl_set1(i) = f_set1(j-1);
        break;
    else
        f_inpl_set1(i) = 1;
    end
end
% set2
for j = 2:length(x_set2)
    if x_ref < x_set2(j-1)
        f_inpl_set2(i) = 0;
        break;
    elseif x_ref >= x_set2(j-1) && x_ref < x_set2(j)
        f_inpl_set2(i) = f_set2(j-1);
        break;
    else
        f_inpl_set2(i) = 1;
    end
end
end

%%% KSI calculation
%vc = 1.36/sqrt(length(data_set2)); % 95% CI
vc = 1.63/sqrt(length(data_set2)); % 99% CI

a_cri = vc*(x_max - x_min);
KSI = 100*sum(abs(f_inpl_set1-f_inpl_set2))*x_sam/a_cri;

%%% OVER calculation
extPnts = zeros(100,1);
overCnt = 0;
for i = 1:length(f_inpl_set1)

    if abs(f_inpl_set1(i)-f_inpl_set2(i))>vc
        overCnt = overCnt + 1;
        extPnts(overCnt) = abs(f_inpl_set1(i)-f_inpl_set2(i));
    end

end

extPnts = extPnts(1:overCnt);
OVER = 100*(sum(extPnts-vc))*x_sam/a_cri;

%% Plotting the hit rate result
f1 = figure(1);
lnWidth = 1.5;
mkSize = 7;
xaxis = 1:31;
lowDist = estKD(xaxis)-estKD_lo(xaxis);
uppDist = estKD_up(xaxis)-estKD(xaxis);
subplot(211)
errorbar(xaxis,estKD(xaxis),lowDist,uppDist,'ks',...
    'LineWidth',lnWidth,'MarkerFaceColor','k','MarkerSize',mkSize)
hold on
plot(xaxis,refKD(xaxis),'ro','MarkerFaceColor','r','MarkerSize',mkSize)
hold off

```

```

xlim([0 32]);ylim([0 1.3])
xlabel('Days'); ylabel('K_D');title('(a) Method 1')
legend('Forecast','Observation')
%print(f1,'-dtiff','-r500','SimulationResult_m1');

```

● EstimationTest_Method2_2012.m

```

%% Estimation Test of Method 2 Using the 2012 Data
% This script tests the estimation performance of the proposed Method 2
% using the 2012 data. The RMSE, MBE, KSI, OVER, and hit rate are observed
% for the performance.

```

```

clear all; close all; clc;

```

```

load Golden_2009_2012_YearlyGroup_ClassGroup_TKCDfor_TKCDobs.mat

```

```

%% Data selection

```

```

yr = 2012;
KD_selYr = eval(strcat('KD_',num2str(yr)));
refKD = KD_selYr;

```

```

TCCDfor_selYr = eval(strcat('TCCDfor_',num2str(yr)));
TCCDfor = TCCDfor_selYr;

```

```

numDysYr = length(KD_selYr(:,1));

```

```

%% Estimating KD

```

```

% Relationship between KD and TCCDobs:
% KDhat_trans = -0.2825*xdata - 0.1979;
% KDhat = (lambda*KDhat_trans+1)^(1/lambda);

```

```

lambda = 2.0906;

```

```

u_TKCDfor = 0.2;
u_a = 0.002908; u_b = 0.001327;
a = -0.2825; b = -0.1979;
sigma = 0.0565;

```

```

TKCDfor = TCCDfor - 0.1148;

```

```

for i = 1:numDysYr
    if TKCDfor(i) < 0
        TKCDfor(i) = 0;
    end
end

```

```

estKD = zeros(numDysYr,1);
estKD_up = zeros(numDysYr,1);
estKD_lo = zeros(numDysYr,1);

```

```

for i = 1:numDysYr
    % estimation

```

```

    if (-0.2825*TKCDfor(i)-0.1979)*lambda > -1
        estKD(i) = (lambda*(-0.2825*TKCDfor(i)-0.1979)+1)^(1/lambda);
    else
        estKD(i) = 0;
    end

    % upper
    if (-0.2825*TKCDfor(i)-0.1979+1.96*sigma)*lambda > -1
        estKD_up(i) = (lambda*(-0.2825*TKCDfor(i)-
0.1979+1.96*sigma)+1)^(1/lambda);
    else
        estKD_up(i) = 0;
    end

    % lower
    if (-0.2825*TKCDfor(i)-0.1979-1.96*sigma)*lambda > -1
        estKD_lo(i) = (lambda*(-0.2825*TKCDfor(i)-0.1979-
1.96*sigma)+1)^(1/lambda);
    else
        estKD_lo(i) = 0;
    end
end

%% Classification by high, mid, and low KD level
numDys_hig = 0; numDys_mid = 0; numDys_low = 0;
idxDys_hig = zeros(numDysYr,1);
idxDys_mid = zeros(numDysYr,1);
idxDys_low = zeros(numDysYr,1);

for i = 1:numDysYr
    if refKD(i) >= 0.6
        numDys_hig = numDys_hig + 1;
        idxDys_hig(numDys_hig) = i;
    elseif refKD(i) < 0.3
        numDys_low = numDys_low + 1;
        idxDys_low(numDys_low) = i;
    else
        numDys_mid = numDys_mid + 1;
        idxDys_mid(numDys_mid) = i;
    end
end

idxDys_hig = idxDys_hig(1:numDys_hig);
idxDys_mid = idxDys_mid(1:numDys_mid);
idxDys_low = idxDys_low(1:numDys_low);

refKD_hig = refKD; refKD_hig = refKD_hig(idxDys_hig(:,1));
refKD_mid = refKD; refKD_mid = refKD_mid(idxDys_mid(:,1));
refKD_low = refKD; refKD_low = refKD_low(idxDys_low(:,1));

estKD_hig = estKD; estKD_hig = estKD_hig(idxDys_hig(:,1));
estKD_mid = estKD; estKD_mid = estKD_mid(idxDys_mid(:,1));
estKD_low = estKD; estKD_low = estKD_low(idxDys_low(:,1));

%% Result analysis for KD

```



```

idxInOut = zeros(numDysYr,3);
idxInOut(idxDys_hig(:,1),1) = 1;
idxInOut(idxDys_mid(:,1),1) = 2;
idxInOut(idxDys_low(:,1),1) = 3;

for i = 1:numDysYr
    if refKD(i)<=estKD_up(i) && refKD(i)>=estKD_lo(i)
        idxInOut(i,2) = 1;
    elseif refKD(i)>estKD_up(i)
        idxInOut(i,2) = 0;
        idxInOut(i,3) = refKD(i) - estKD_up(i);
    else
        idxInOut(i,2) = 0;
        idxInOut(i,3) = refKD(i) - estKD_lo(i);
    end
end

numIn = length(idxInOut(idxInOut(:,2)==1,2));

idxInOut_hig = idxInOut(idxInOut(:,1)==1,:);
numIn_hig = length(idxInOut_hig(idxInOut_hig(:,2)==1,2));

idxInOut_mid = idxInOut(idxInOut(:,1)==2,:);
numIn_mid = length(idxInOut_mid(idxInOut_mid(:,2)==1,2));

idxInOut_low = idxInOut(idxInOut(:,1)==3,:);
numIn_low = length(idxInOut_low(idxInOut_low(:,2)==1,2));

if sum(idxInOut(idxInOut(:,2)==0,3)) == 0
    MAOutDist = 0;
else
    MAOutDist = mean(abs(idxInOut(idxInOut(:,2)==0,3)));
    MBOutDist = mean(idxInOut(idxInOut(:,2)==0,3));
end

if sum(idxInOut_hig(idxInOut_hig(:,2)==0,3)) == 0
    MAOutDist_hig = 0;
    MBOutDist_hig = 0;
else
    MAOutDist_hig = mean(abs(idxInOut_hig(idxInOut_hig(:,2)==0,3)));
    MBOutDist_hig = mean(idxInOut_hig(idxInOut_hig(:,2)==0,3));
end

if sum(idxInOut_mid(idxInOut_mid(:,2)==0,3)) == 0
    MAOutDist_mid = 0;
    MBOutDist_mid = 0;
else
    MAOutDist_mid = mean(abs(idxInOut_mid(idxInOut_mid(:,2)==0,3)));
    MBOutDist_mid = mean(idxInOut_mid(idxInOut_mid(:,2)==0,3));
end

if sum(idxInOut_low(idxInOut_low(:,2)==0,3)) == 0
    MAOutDist_low = 0;
    MBOutDist_low = 0;
else

```

```

        MAOutDist_low = mean(abs(idxInOut_low(idxInOut_low(:,2)==0,3)));
        MBOutDist_low = mean(idxInOut_low(idxInOut_low(:,2)==0,3));
end

perIn = numIn/numDysYr * 100;
perIn_hig = numIn_hig/numDys_hig * 100;
perIn_mid = numIn_mid/numDys_mid * 100;
perIn_low = numIn_low/numDys_low * 100;

%% RMSE and MBE for KD
RMSE_KD = sqrt(mean((refKD - estKD).^2));
MBE_KD = mean(estKD - refKD);
MAE_KD = mean(abs(refKD - estKD));

RMSE_KD_hig = sqrt(mean((refKD_hig - estKD_hig).^2));
MBE_KD_hig = mean(refKD_hig - estKD_hig);
MAE_KD_hig = mean(abs(refKD_hig - estKD_hig));

RMSE_KD_mid = sqrt(mean((refKD_mid - estKD_mid).^2));
MBE_KD_mid = mean(refKD_mid - estKD_mid);
MAE_KD_mid = mean(abs(refKD_mid - estKD_mid));

RMSE_KD_low = sqrt(mean((refKD_low - estKD_low).^2));
MBE_KD_low = mean(refKD_low - estKD_low);
MAE_KD_low = mean(abs(refKD_low - estKD_low));

rRMSE_KD = sqrt(mean((refKD - estKD).^2)/mean(refKD.^2));
rRMSE_KD_hig = sqrt(mean((refKD_hig - estKD_hig).^2)/mean(refKD_hig.^2));
rRMSE_KD_mid = sqrt(mean((refKD_mid - estKD_mid).^2)/mean(refKD_mid.^2));
rRMSE_KD_low = sqrt(mean((refKD_low - estKD_low).^2)/mean(refKD_low.^2));

rMBE_KD = mean(estKD-refKD)/mean(refKD);
rMBE_KD_hig = mean(refKD_hig - estKD_hig)/mean(refKD_hig);
rMBE_KD_mid = mean(refKD_mid - estKD_mid)/mean(refKD_mid);
rMBE_KD_low = mean(refKD_low - estKD_low)/mean(refKD_low);

%% RMSE and MBE for TKCD
TCCDobs_selyr = eval(strcat('TCCDobs_', num2str(yr)));
refTKCD = TCCDobs_selyr;
estTKCD = TKCDfor;

refTKCD_hig = refTKCD; refTKCD_hig = refTKCD_hig(idxDys_hig(:,1));
refTKCD_mid = refTKCD; refTKCD_mid = refTKCD_mid(idxDys_mid(:,1));
refTKCD_low = refTKCD; refTKCD_low = refTKCD_low(idxDys_low(:,1));

estTKCD_hig = estTKCD; estTKCD_hig = estTKCD_hig(idxDys_hig(:,1));
estTKCD_mid = estTKCD; estTKCD_mid = estTKCD_mid(idxDys_mid(:,1));
estTKCD_low = estTKCD; estTKCD_low = estTKCD_low(idxDys_low(:,1));

RMSE_TKCD = sqrt(mean((refTKCD - estTKCD).^2));
RMSE_TKCD_hig = sqrt(mean((refTKCD_hig - estTKCD_hig).^2));
RMSE_TKCD_mid = sqrt(mean((refTKCD_mid - estTKCD_mid).^2));
RMSE_TKCD_low = sqrt(mean((refTKCD_low - estTKCD_low).^2));

```

```

rRMSE_TKCD = sqrt(mean((refTKCD - estTKCD).^2)/mean(refTKCD.^2));
rRMSE_TKCD_hig = sqrt(mean((refTKCD_hig - estTKCD_hig).^2)/mean(refTKCD_hig.^2));
rRMSE_TKCD_mid = sqrt(mean((refTKCD_mid - estTKCD_mid).^2)/mean(refTKCD_mid.^2));
rRMSE_TKCD_low = sqrt(mean((refTKCD_low - estTKCD_low).^2)/mean(refTKCD_low.^2));

MBE_TKCD = mean(estTKCD - refTKCD);
MBE_TKCD_hig = mean(refTKCD_hig - estTKCD_hig);
MBE_TKCD_mid = mean(refTKCD_mid - estTKCD_mid);
MBE_TKCD_low = mean(refTKCD_low - estTKCD_low);

rMBE_TKCD = mean(estTKCD-refTKCD)/mean(refTKCD);
rMBE_TKCD_hig = mean(refTKCD_hig - estTKCD_hig)/mean(refTKCD_hig);
rMBE_TKCD_mid = mean(refTKCD_mid - estTKCD_mid)/mean(refTKCD_mid);
rMBE_TKCD_low = mean(refTKCD_low - estTKCD_low)/mean(refTKCD_low);

%% KSI and OVER calculation
%%% Interpolation of 100 points
data_set1 = KD_2012;
data_set2 = estKD;

[f_set1,x_set1] = ecdf(data_set1);
[f_set2,x_set2] = ecdf(data_set2);

x_min_set1 = min(x_set1);
x_min_set2 = min(x_set2);

x_max_set1 = max(x_set1);
x_max_set2 = max(x_set2);

if x_min_set1 < x_min_set2
    x_min = x_min_set1;
else
    x_min = x_min_set2;
end
if x_max_set1 > x_max_set2
    x_max = x_max_set1;
else
    x_max = x_max_set2;
end

x_sam = (x_max - x_min) / 100;

xi = x_min:x_sam:x_max-x_sam;

f_inpl_set1 = zeros(100,1);
f_inpl_set2 = zeros(100,1);

for i = 1:100
    x_ref = xi(i);
    % set1
    for j = 2:length(x_set1)
        if x_ref < x_set1(j-1)

```

```

        f_inpl_set1(i) = 0;
        break;
    elseif x_ref >= x_set1(j-1) && x_ref < x_set1(j)
        f_inpl_set1(i) = f_set1(j-1);
        break;
    else
        f_inpl_set1(i) = 1;
    end
end
% set2
for j = 2:length(x_set2)
    if x_ref < x_set2(j-1)
        f_inpl_set2(i) = 0;
        break;
    elseif x_ref >= x_set2(j-1) && x_ref < x_set2(j)
        f_inpl_set2(i) = f_set2(j-1);
        break;
    else
        f_inpl_set2(i) = 1;
    end
end
end

%%% KSI calculation
%vc = 1.36/sqrt(length(data_set2)); % 95% CI
vc = 1.63/sqrt(length(data_set2)); % 99% CI

a_cri = vc*(x_max - x_min);
KSI = 100*sum(abs(f_inpl_set1-f_inpl_set2))*x_sam/a_cri;

%%% OVER calculation
extPnts = zeros(100,1);
overCnt = 0;
for i = 1:length(f_inpl_set1)

    if abs(f_inpl_set1(i)-f_inpl_set2(i))>vc
        overCnt = overCnt + 1;
        extPnts(overCnt) = abs(f_inpl_set1(i)-f_inpl_set2(i));
    end

end

extPnts = extPnts(1:overCnt);
OVER = 100*(sum(extPnts-vc))*x_sam/a_cri;

%% Plotting the hit rate result
f1 = figure(1);
lnWidth = 1.5;
mkSize = 7;
xaxis = 1:31;
lowDist = estKD(xaxis)-estKD_lo(xaxis);
uppDist = estKD_up(xaxis)-estKD(xaxis);
subplot(211)
errorbar(xaxis,estKD(xaxis),lowDist,uppDist,'ks',...
    'LineWidth',lnWidth,'MarkerFaceColor','k','MarkerSize',mkSize)
hold on
plot(xaxis,refKD(xaxis),'ro','MarkerFaceColor','r','MarkerSize',mkSize)

```

```

hold off
xlim([0 32]);ylim([0 1.3])
xlabel('Days'); ylabel('K_D');title('(b) Method 2')
legend('Forecast','Observation')
%print(f1,'-dtiff','-r500','SimulationResult_m2');

```

● EstimationTest_Method3_2012.m

```

%% Estimation Test of Method 3 Using the 2012 Data
% This script tests the estimation performance of the proposed Method 3
% using the 2012 data. The RMSE, MBE, KSI, OVER, and hit rate are observed
% for the performance.

clear all; close all; clc;

load Golden_2009_2012_YearlyGroup_ClassGroup_TKCDfor_TKCDobs.mat

%% Data selection
yr = 2012;
KD_selYr = eval(strcat('KD_',num2str(yr)));
refKD = KD_selYr;

TCCDfor_selYr = eval(strcat('TCCDfor_',num2str(yr)));
TCCDfor = TCCDfor_selYr;

numDysYr = length(KD_selYr(:,1));

%% Estimating KD
% Relationship between KD and TKCDobs:
% KDhat_trans = -0.2825*xdata - 0.1979;
% KDhat = (lambda*KDhat_trans+1)^(1/lambda);

lambda = 2.0906;

u_TKCDfor = 0.2;
u_a = 0.002908; u_b = 0.001327;
a = -0.2825; b = -0.1979;
sigma = 0.0565;

TKCDfor = zeros(numDysYr,1);
for i = 1:numDysYr
    if TCCDfor(i) <= 0.4106 % high
        TKCDfor(i) = TCCDfor(i) - 0.1986;
    elseif TCCDfor(i) > 0.8559 % low
        TKCDfor(i) = TCCDfor(i) - 0.0513;
    else % medium
        TKCDfor(i) = TCCDfor(i) - 0.0008;
    end

    if TKCDfor(i) < 0
        TKCDfor(i) = 0;
    end
end

```

```

end

estKD = zeros(numDysYr,1);
estKD_up = zeros(numDysYr,1);
estKD_lo = zeros(numDysYr,1);

for i = 1:numDysYr
    % estimation
    if (-0.2825*TKCDfor(i)-0.1979)*lambda > -1
        estKD(i) = (lambda*(-0.2825*TKCDfor(i)-0.1979)+1)^(1/lambda);
    else
        estKD(i) = 0;
    end

    % upper
    if (-0.2825*TKCDfor(i)-0.1979+1.96*sigma)*lambda > -1
        estKD_up(i) = (lambda*(-0.2825*TKCDfor(i)-
0.1979+1.96*sigma)+1)^(1/lambda);
    else
        estKD_up(i) = 0;
    end

    % lower
    if (-0.2825*TKCDfor(i)-0.1979-1.96*sigma)*lambda > -1
        estKD_lo(i) = (lambda*(-0.2825*TKCDfor(i)-0.1979-
1.96*sigma)+1)^(1/lambda);
    else
        estKD_lo(i) = 0;
    end
end

%% Classification by high, mid, and low KD level
numDys_hig = 0; numDys_mid = 0; numDys_low = 0;
idxDys_hig = zeros(numDysYr,1);
idxDys_mid = zeros(numDysYr,1);
idxDys_low = zeros(numDysYr,1);

for i = 1:numDysYr
    if refKD(i) >= 0.6
        numDys_hig = numDys_hig + 1;
        idxDys_hig(numDys_hig) = i;
    elseif refKD(i) < 0.3
        numDys_low = numDys_low + 1;
        idxDys_low(numDys_low) = i;
    else
        numDys_mid = numDys_mid + 1;
        idxDys_mid(numDys_mid) = i;
    end
end

idxDys_hig = idxDys_hig(1:numDys_hig);
idxDys_mid = idxDys_mid(1:numDys_mid);
idxDys_low = idxDys_low(1:numDys_low);

refKD_hig = refKD; refKD_hig = refKD_hig(idxDys_hig(:,1));

```

```

refKD_mid = refKD; refKD_mid = refKD_mid(idxDys_mid(:,1));
refKD_low = refKD; refKD_low = refKD_low(idxDys_low(:,1));

estKD_hig = estKD; estKD_hig = estKD_hig(idxDys_hig(:,1));
estKD_mid = estKD; estKD_mid = estKD_mid(idxDys_mid(:,1));
estKD_low = estKD; estKD_low = estKD_low(idxDys_low(:,1));

%% Result analysis for KD
idxInOut = zeros(numDysYr,3);
idxInOut(idxDys_hig(:,1),1) = 1;
idxInOut(idxDys_mid(:,1),1) = 2;
idxInOut(idxDys_low(:,1),1) = 3;

for i = 1:numDysYr
    if refKD(i)<=estKD_up(i) && refKD(i)>=estKD_lo(i)
        idxInOut(i,2) = 1;
    elseif refKD(i)>estKD_up(i)
        idxInOut(i,2) = 0;
        idxInOut(i,3) = refKD(i) - estKD_up(i);
    else
        idxInOut(i,2) = 0;
        idxInOut(i,3) = refKD(i) - estKD_lo(i);
    end
end

numIn = length(idxInOut(idxInOut(:,2)==1,2));

idxInOut_hig = idxInOut(idxInOut(:,1)==1,:);
numIn_hig = length(idxInOut_hig(idxInOut_hig(:,2)==1,2));

idxInOut_mid = idxInOut(idxInOut(:,1)==2,:);
numIn_mid = length(idxInOut_mid(idxInOut_mid(:,2)==1,2));

idxInOut_low = idxInOut(idxInOut(:,1)==3,:);
numIn_low = length(idxInOut_low(idxInOut_low(:,2)==1,2));

if sum(idxInOut(idxInOut(:,2)==0,3)) == 0
    MAOutDist = 0;
else
    MAOutDist = mean(abs(idxInOut(idxInOut(:,2)==0,3)));
    MBOutDist = mean(idxInOut(idxInOut(:,2)==0,3));
end

if sum(idxInOut_hig(idxInOut_hig(:,2)==0,3)) == 0
    MAOutDist_hig = 0;
    MBOutDist_hig = 0;
else
    MAOutDist_hig = mean(abs(idxInOut_hig(idxInOut_hig(:,2)==0,3)));
    MBOutDist_hig = mean(idxInOut_hig(idxInOut_hig(:,2)==0,3));
end

if sum(idxInOut_mid(idxInOut_mid(:,2)==0,3)) == 0
    MAOutDist_mid = 0;
    MBOutDist_mid = 0;
else

```

```

        MAOutDist_mid = mean(abs(idxInOut_mid(idxInOut_mid(:,2)==0,3)));
        MBOutDist_mid = mean(idxInOut_mid(idxInOut_mid(:,2)==0,3));
end

if sum(idxInOut_low(idxInOut_low(:,2)==0,3)) == 0
    MAOutDist_low = 0;
    MBOutDist_low = 0;
else
    MAOutDist_low = mean(abs(idxInOut_low(idxInOut_low(:,2)==0,3)));
    MBOutDist_low = mean(idxInOut_low(idxInOut_low(:,2)==0,3));
end

perIn = numIn/numDysYr * 100;
perIn_hig = numIn_hig/numDys_hig * 100;
perIn_mid = numIn_mid/numDys_mid * 100;
perIn_low = numIn_low/numDys_low * 100;

%% RMSE and MBE for KD
RMSE_KD = sqrt(mean((refKD - estKD).^2));
MBE_KD = mean(estKD - refKD);
MAE_KD = mean(abs(refKD - estKD));

RMSE_KD_hig = sqrt(mean((refKD_hig - estKD_hig).^2));
MBE_KD_hig = mean(refKD_hig - estKD_hig);
MAE_KD_hig = mean(abs(refKD_hig - estKD_hig));

RMSE_KD_mid = sqrt(mean((refKD_mid - estKD_mid).^2));
MBE_KD_mid = mean(refKD_mid - estKD_mid);
MAE_KD_mid = mean(abs(refKD_mid - estKD_mid));

RMSE_KD_low = sqrt(mean((refKD_low - estKD_low).^2));
MBE_KD_low = mean(refKD_low - estKD_low);
MAE_KD_low = mean(abs(refKD_low - estKD_low));

rRMSE_KD = sqrt(mean((refKD - estKD).^2)/mean(refKD.^2));
rRMSE_KD_hig = sqrt(mean((refKD_hig - estKD_hig).^2)/mean(refKD_hig.^2));
rRMSE_KD_mid = sqrt(mean((refKD_mid - estKD_mid).^2)/mean(refKD_mid.^2));
rRMSE_KD_low = sqrt(mean((refKD_low - estKD_low).^2)/mean(refKD_low.^2));

rMBE_KD = mean(estKD-refKD)/mean(refKD);
rMBE_KD_hig = mean(refKD_hig - estKD_hig)/mean(refKD_hig);
rMBE_KD_mid = mean(refKD_mid - estKD_mid)/mean(refKD_mid);
rMBE_KD_low = mean(refKD_low - estKD_low)/mean(refKD_low);

%% RMSE and MBE for TKCD
TCCDobs_selYr = eval(strcat('TCCDobs_', num2str(yr)));
refTKCD = TCCDobs_selYr;
estTKCD = TKCDfor;

refTKCD_hig = refTKCD; refTKCD_hig = refTKCD_hig(idxDys_hig(:,1));
refTKCD_mid = refTKCD; refTKCD_mid = refTKCD_mid(idxDys_mid(:,1));
refTKCD_low = refTKCD; refTKCD_low = refTKCD_low(idxDys_low(:,1));

estTKCD_hig = estTKCD; estTKCD_hig = estTKCD_hig(idxDys_hig(:,1));
estTKCD_mid = estTKCD; estTKCD_mid = estTKCD_mid(idxDys_mid(:,1));

```



```

estTKCD_low = estTKCD; estTKCD_low = estTKCD_low(idxDys_low(:,1));

RMSE_TKCD = sqrt(mean((refTKCD - estTKCD).^2));
RMSE_TKCD_hig = sqrt(mean((refTKCD_hig - estTKCD_hig).^2));
RMSE_TKCD_mid = sqrt(mean((refTKCD_mid - estTKCD_mid).^2));
RMSE_TKCD_low = sqrt(mean((refTKCD_low - estTKCD_low).^2));

rRMSE_TKCD = sqrt(mean((refTKCD - estTKCD).^2)/mean(refTKCD.^2));
rRMSE_TKCD_hig = sqrt(mean((refTKCD_hig - estTKCD_hig).^2)/mean(refTKCD_hig.^2));
rRMSE_TKCD_mid = sqrt(mean((refTKCD_mid - estTKCD_mid).^2)/mean(refTKCD_mid.^2));
rRMSE_TKCD_low = sqrt(mean((refTKCD_low - estTKCD_low).^2)/mean(refTKCD_low.^2));

MBE_TKCD = mean(estTKCD - refTKCD);
MBE_TKCD_hig = mean(refTKCD_hig - estTKCD_hig);
MBE_TKCD_mid = mean(refTKCD_mid - estTKCD_mid);
MBE_TKCD_low = mean(refTKCD_low - estTKCD_low);

rMBE_TKCD = mean(estTKCD-refTKCD)/mean(refTKCD);
rMBE_TKCD_hig = mean(refTKCD_hig - estTKCD_hig)/mean(refTKCD_hig);
rMBE_TKCD_mid = mean(refTKCD_mid - estTKCD_mid)/mean(refTKCD_mid);
rMBE_TKCD_low = mean(refTKCD_low - estTKCD_low)/mean(refTKCD_low);

%% KSI and OVER calculation
%%% Interpolation of 100 points
data_set1 = KD_2012;
data_set2 = estKD;

[f_set1,x_set1] = ecdf(data_set1);
[f_set2,x_set2] = ecdf(data_set2);

x_min_set1 = min(x_set1);
x_min_set2 = min(x_set2);

x_max_set1 = max(x_set1);
x_max_set2 = max(x_set2);

if x_min_set1 < x_min_set2
    x_min = x_min_set1;
else
    x_min = x_min_set2;
end
if x_max_set1 > x_max_set2
    x_max = x_max_set1;
else
    x_max = x_max_set2;
end

x_sam = (x_max - x_min) / 100;

xi = x_min:x_sam:x_max-x_sam;

```

```

f_inpl_set1 = zeros(100,1);
f_inpl_set2 = zeros(100,1);

for i = 1:100
    x_ref = xi(i);
    % set1
    for j = 2:length(x_set1)
        if x_ref < x_set1(j-1)
            f_inpl_set1(i) = 0;
            break;
        elseif x_ref >= x_set1(j-1) && x_ref < x_set1(j)
            f_inpl_set1(i) = f_set1(j-1);
            break;
        else
            f_inpl_set1(i) = 1;
        end
    end
    % set2
    for j = 2:length(x_set2)
        if x_ref < x_set2(j-1)
            f_inpl_set2(i) = 0;
            break;
        elseif x_ref >= x_set2(j-1) && x_ref < x_set2(j)
            f_inpl_set2(i) = f_set2(j-1);
            break;
        else
            f_inpl_set2(i) = 1;
        end
    end
end

%%% KSI calculation
%vc = 1.36/sqrt(length(data_set2)); % 95% CI
vc = 1.63/sqrt(length(data_set2)); % 99% CI

a_cri = vc*(x_max - x_min);
KSI = 100*sum(abs(f_inpl_set1-f_inpl_set2))*x_sam/a_cri;

%%% OVER calculation
extPnts = zeros(100,1);
overCnt = 0;
for i = 1:length(f_inpl_set1)

    if abs(f_inpl_set1(i)-f_inpl_set2(i))>vc
        overCnt = overCnt + 1;
        extPnts(overCnt) = abs(f_inpl_set1(i)-f_inpl_set2(i));
    end

end

extPnts = extPnts(1:overCnt);
OVER = 100*(sum(extPnts-vc))*x_sam/a_cri;

%% Plotting the hit rate result
f1 = figure(1);
lnWidth = 1.5;

```

```

mkSize = 7;
xaxis = 1:31;
lowDist = estKD(xaxis)-estKD_lo(xaxis);
uppDist = estKD_up(xaxis)-estKD(xaxis);
subplot(211)
errorbar(xaxis,estKD(xaxis),lowDist,uppDist,'ks',...
    'LineWidth',lnWidth,'MarkerFaceColor','k','MarkerSize',mkSize)
hold on
plot(xaxis,refKD(xaxis),'ro','MarkerFaceColor','r','MarkerSize',mkSize)
hold off
xlim([0 32]);ylim([0 1.3])
xlabel('Days'); ylabel('K_D');title('(c) Method 3')
legend('Forecast','Observation')
%print(f1,'-dtiff','-r500','SimulationResult_m3');

```

● EstimationTest_Method4_2012.m

```

%% Estimation Test of Method 4 Using the 2012 Data
% This script tests the estimation performance of the proposed Method 4
% using the 2012 data. The RMSE, MBE, KSI, OVER, and hit rate are observed
% for the performance.

```

```

clear all; close all; clc;

```

```

load Golden_2009_2012_YearlyGroup_ClassGroup_TKCDfor_TKCDobs.mat

```

```

%% Data selection

```

```

yr = 2012;
KD_selYr = eval(strcat('KD_',num2str(yr)));
refKD = KD_selYr;

```

```

TKCDfor_selYr = eval(strcat('TKCDfor_',num2str(yr)));
TKCDfor = TKCDfor_selYr;

```

```

numDysYr = length(KD_selYr(:,1));

```

```

%% Estimating KD

```

```

% Relationship between KD and TKCDobs:
% KDhat_trans = -0.2825*xdata - 0.1979;
% KDhat = (lambda*KDhat_trans+1)^(1/lambda);

```

```

lambda = 2.0906;
u_TKCDfor = 0.2;
u_a = 0.002908; u_b = 0.001327;
a = -0.2825; b = -0.1979;
sigma = 0.0565;

```

```

TKCDfor = zeros(numDysYr,1);
for i = 1:numDysYr
    if TKCDfor(i) <= 0.4106 % high
        TKCDfor(i) = TKCDfor(i) - 0.1148;
    elseif TKCDfor(i) > 0.8559 % low

```

```

        TKCDfor(i) = TCCDfor(i) - 0.0513;
    else % medium
        TKCDfor(i) = TCCDfor(i) - 0.0008;
    end

    if TKCDfor(i) < 0
        TKCDfor(i) = 0;
    end
end

estKD = zeros(numDysYr,1);
estKD_up = zeros(numDysYr,1);
estKD_lo = zeros(numDysYr,1);

for i = 1:numDysYr
    % estimation
    if (-0.2825*TKCDfor(i)-0.1979)*lambda > -1
        estKD(i) = (lambda*(-0.2825*TKCDfor(i)-0.1979)+1)^(1/lambda);
    else
        estKD(i) = 0;
    end

    % upper
    if (-0.2825*TKCDfor(i)-0.1979+1.96*sigma)*lambda > -1
        estKD_up(i) = (lambda*(-0.2825*TKCDfor(i)-
0.1979+1.96*sigma)+1)^(1/lambda);
    else
        estKD_up(i) = 0;
    end

    % lower
    if (-0.2825*TKCDfor(i)-0.1979-1.96*sigma)*lambda > -1
        estKD_lo(i) = (lambda*(-0.2825*TKCDfor(i)-0.1979-
1.96*sigma)+1)^(1/lambda);
    else
        estKD_lo(i) = 0;
    end
end

%% Classification by high, mid, and low KD level
numDys_hig = 0; numDys_mid = 0; numDys_low = 0;
idxDys_hig = zeros(numDysYr,1);
idxDys_mid = zeros(numDysYr,1);
idxDys_low = zeros(numDysYr,1);

for i = 1:numDysYr
    if refKD(i) >= 0.6
        numDys_hig = numDys_hig + 1;
        idxDys_hig(numDys_hig) = i;
    elseif refKD(i) < 0.3
        numDys_low = numDys_low + 1;
        idxDys_low(numDys_low) = i;
    else
        numDys_mid = numDys_mid + 1;
        idxDys_mid(numDys_mid) = i;
    end
end

```

```

end
end

idxDys_hig = idxDys_hig(1:numDys_hig);
idxDys_mid = idxDys_mid(1:numDys_mid);
idxDys_low = idxDys_low(1:numDys_low);

refKD_hig = refKD; refKD_hig = refKD_hig(idxDys_hig(:,1));
refKD_mid = refKD; refKD_mid = refKD_mid(idxDys_mid(:,1));
refKD_low = refKD; refKD_low = refKD_low(idxDys_low(:,1));

estKD_hig = estKD; estKD_hig = estKD_hig(idxDys_hig(:,1));
estKD_mid = estKD; estKD_mid = estKD_mid(idxDys_mid(:,1));
estKD_low = estKD; estKD_low = estKD_low(idxDys_low(:,1));

%% Result analysis for KD
idxInOut = zeros(numDysYr,3);
idxInOut(idxDys_hig(:,1),1) = 1;
idxInOut(idxDys_mid(:,1),1) = 2;
idxInOut(idxDys_low(:,1),1) = 3;

for i = 1:numDysYr
    if refKD(i)<=estKD_up(i) && refKD(i)>=estKD_lo(i)
        idxInOut(i,2) = 1;
    elseif refKD(i)>estKD_up(i)
        idxInOut(i,2) = 0;
        idxInOut(i,3) = refKD(i) - estKD_up(i);
    else
        idxInOut(i,2) = 0;
        idxInOut(i,3) = refKD(i) - estKD_lo(i);
    end
end

numIn = length(idxInOut(idxInOut(:,2)==1,2));

idxInOut_hig = idxInOut(idxInOut(:,1)==1,:);
numIn_hig = length(idxInOut_hig(idxInOut_hig(:,2)==1,2));

idxInOut_mid = idxInOut(idxInOut(:,1)==2,:);
numIn_mid = length(idxInOut_mid(idxInOut_mid(:,2)==1,2));

idxInOut_low = idxInOut(idxInOut(:,1)==3,:);
numIn_low = length(idxInOut_low(idxInOut_low(:,2)==1,2));

if sum(idxInOut(idxInOut(:,2)==0,3)) == 0
    MAOutDist = 0;
else
    MAOutDist = mean(abs(idxInOut(idxInOut(:,2)==0,3)));
    MBOutDist = mean(idxInOut(idxInOut(:,2)==0,3));
end

if sum(idxInOut_hig(idxInOut_hig(:,2)==0,3)) == 0
    MAOutDist_hig = 0;
    MBOutDist_hig = 0;
else

```

```

        MAOutDist_hig = mean(abs(idxInOut_hig(idxInOut_hig(:,2)==0,3)));
        MBOutDist_hig = mean(idxInOut_hig(idxInOut_hig(:,2)==0,3));
end

if sum(idxInOut_mid(idxInOut_mid(:,2)==0,3)) == 0
    MAOutDist_mid = 0;
    MBOutDist_mid = 0;
else
    MAOutDist_mid = mean(abs(idxInOut_mid(idxInOut_mid(:,2)==0,3)));
    MBOutDist_mid = mean(idxInOut_mid(idxInOut_mid(:,2)==0,3));
end

if sum(idxInOut_low(idxInOut_low(:,2)==0,3)) == 0
    MAOutDist_low = 0;
    MBOutDist_low = 0;
else
    MAOutDist_low = mean(abs(idxInOut_low(idxInOut_low(:,2)==0,3)));
    MBOutDist_low = mean(idxInOut_low(idxInOut_low(:,2)==0,3));
end

perIn = numIn/numDysYr * 100;
perIn_hig = numIn_hig/numDys_hig * 100;
perIn_mid = numIn_mid/numDys_mid * 100;
perIn_low = numIn_low/numDys_low * 100;

%% RMSE and MBE for KD
RMSE_KD = sqrt(mean((refKD - estKD).^2));
MBE_KD = mean(estKD - refKD);
MAE_KD = mean(abs(refKD - estKD));

RMSE_KD_hig = sqrt(mean((refKD_hig - estKD_hig).^2));
MBE_KD_hig = mean(refKD_hig - estKD_hig);
MAE_KD_hig = mean(abs(refKD_hig - estKD_hig));

RMSE_KD_mid = sqrt(mean((refKD_mid - estKD_mid).^2));
MBE_KD_mid = mean(refKD_mid - estKD_mid);
MAE_KD_mid = mean(abs(refKD_mid - estKD_mid));

RMSE_KD_low = sqrt(mean((refKD_low - estKD_low).^2));
MBE_KD_low = mean(refKD_low - estKD_low);
MAE_KD_low = mean(abs(refKD_low - estKD_low));

rRMSE_KD = sqrt(mean((refKD - estKD).^2)/mean(refKD.^2));
rRMSE_KD_hig = sqrt(mean((refKD_hig - estKD_hig).^2)/mean(refKD_hig.^2));
rRMSE_KD_mid = sqrt(mean((refKD_mid - estKD_mid).^2)/mean(refKD_mid.^2));
rRMSE_KD_low = sqrt(mean((refKD_low - estKD_low).^2)/mean(refKD_low.^2));

rMBE_KD = mean(estKD-refKD)/mean(refKD);
rMBE_KD_hig = mean(refKD_hig - estKD_hig)/mean(refKD_hig);
rMBE_KD_mid = mean(refKD_mid - estKD_mid)/mean(refKD_mid);
rMBE_KD_low = mean(refKD_low - estKD_low)/mean(refKD_low);

%% RMSE and MBE for TKCD
TCCDobs_selYr = eval(strcat('TCCDobs_', num2str(yr)));
refTKCD = TCCDobs_selYr;

```

```

estTKCD = TKCDfor;

refTKCD_hig = refTKCD; refTKCD_hig = refTKCD_hig(idxDys_hig(:,1));
refTKCD_mid = refTKCD; refTKCD_mid = refTKCD_mid(idxDys_mid(:,1));
refTKCD_low = refTKCD; refTKCD_low = refTKCD_low(idxDys_low(:,1));

estTKCD_hig = estTKCD; estTKCD_hig = estTKCD_hig(idxDys_hig(:,1));
estTKCD_mid = estTKCD; estTKCD_mid = estTKCD_mid(idxDys_mid(:,1));
estTKCD_low = estTKCD; estTKCD_low = estTKCD_low(idxDys_low(:,1));

RMSE_TKCD = sqrt(mean((refTKCD - estTKCD).^2));
RMSE_TKCD_hig = sqrt(mean((refTKCD_hig - estTKCD_hig).^2));
RMSE_TKCD_mid = sqrt(mean((refTKCD_mid - estTKCD_mid).^2));
RMSE_TKCD_low = sqrt(mean((refTKCD_low - estTKCD_low).^2));

rRMSE_TKCD = sqrt(mean((refTKCD - estTKCD).^2)/mean(refTKCD.^2));
rRMSE_TKCD_hig = sqrt(mean((refTKCD_hig - estTKCD_hig).^2)/mean(refTKCD_hig.^2));
rRMSE_TKCD_mid = sqrt(mean((refTKCD_mid - estTKCD_mid).^2)/mean(refTKCD_mid.^2));
rRMSE_TKCD_low = sqrt(mean((refTKCD_low - estTKCD_low).^2)/mean(refTKCD_low.^2));

MBE_TKCD = mean(estTKCD - refTKCD);
MBE_TKCD_hig = mean(refTKCD_hig - estTKCD_hig);
MBE_TKCD_mid = mean(refTKCD_mid - estTKCD_mid);
MBE_TKCD_low = mean(refTKCD_low - estTKCD_low);

rMBE_TKCD = mean(estTKCD-refTKCD)/mean(refTKCD);
rMBE_TKCD_hig = mean(refTKCD_hig - estTKCD_hig)/mean(refTKCD_hig);
rMBE_TKCD_mid = mean(refTKCD_mid - estTKCD_mid)/mean(refTKCD_mid);
rMBE_TKCD_low = mean(refTKCD_low - estTKCD_low)/mean(refTKCD_low);

%% KSI and OVER calculation
%%% Interpolation of 100 points
data_set1 = KD_2012;
data_set2 = estKD;

[f_set1,x_set1] = ecdf(data_set1);
[f_set2,x_set2] = ecdf(data_set2);

x_min_set1 = min(x_set1);
x_min_set2 = min(x_set2);

x_max_set1 = max(x_set1);
x_max_set2 = max(x_set2);

if x_min_set1 < x_min_set2
    x_min = x_min_set1;
else
    x_min = x_min_set2;
end
if x_max_set1 > x_max_set2
    x_max = x_max_set1;
else

```

```

        x_max = x_max_set2;
    end

    %x_min = min(x_set2); x_max = max(x_set2);

    x_sam = (x_max - x_min) / 100;

    xi = x_min:x_sam:x_max-x_sam;

    f_inpl_set1 = zeros(100,1);
    f_inpl_set2 = zeros(100,1);

    for i = 1:100
        x_ref = xi(i);
        % set1
        for j = 2:length(x_set1)
            if x_ref < x_set1(j-1)
                f_inpl_set1(i) = 0;
                break;
            elseif x_ref >= x_set1(j-1) && x_ref < x_set1(j)
                f_inpl_set1(i) = f_set1(j-1);
                break;
            else
                f_inpl_set1(i) = 1;
            end
        end
        % set2
        for j = 2:length(x_set2)
            if x_ref < x_set2(j-1)
                f_inpl_set2(i) = 0;
                break;
            elseif x_ref >= x_set2(j-1) && x_ref < x_set2(j)
                f_inpl_set2(i) = f_set2(j-1);
                break;
            else
                f_inpl_set2(i) = 1;
            end
        end
    end

    end

    %% KSI calculation
    %vc = 1.36/sqrt(length(data_set2)); % 95% CI
    vc = 1.63/sqrt(length(data_set2)); % 99% CI

    a_cri = vc*(x_max - x_min);
    KSI = 100*sum(abs(f_inpl_set1-f_inpl_set2))*x_sam/a_cri;

    %% OVER calculation
    extPnts = zeros(100,1);
    overCnt = 0;
    for i = 1:length(f_inpl_set1)

        if abs(f_inpl_set1(i)-f_inpl_set2(i))>vc
            overCnt = overCnt + 1;
            extPnts(overCnt) = abs(f_inpl_set1(i)-f_inpl_set2(i));
        end
    end

```



```

        end

end
extPnts = extPnts(1:overCnt);
OVER = 100*(sum(extPnts-vc))*x_sam/a_cri;

%% Plotting the hit rate result
f1 = figure(1);
lnWidth = 1.5;
mkSize = 7;
xaxis = 1:31;
lowDist = estKD(xaxis)-estKD_lo(xaxis);
uppDist = estKD_up(xaxis)-estKD(xaxis);
subplot(211)
errorbar(xaxis,estKD(xaxis),lowDist,uppDist,'ks',...
    'LineWidth',lnWidth,'MarkerFaceColor','k','MarkerSize',mkSize)
hold on
plot(xaxis,refKD(xaxis),'ro','MarkerFaceColor','r','MarkerSize',mkSize)
hold off
xlim([0 32]);ylim([0 1.3])
xlabel('Days'); ylabel('K_D');title('(d) Method 4')
legend('Forecast','Observation')
%print(f1,'-dtiff','-r500','SimulationResult_m4');

```

● EstimationTest_Method5_2012.m

```

%% Estimation Test of Method 5 Using the 2012 Data
% This script tests the estimation performance of the proposed Method 5
% using the 2012 data. The RMSE, MBE, KSI, OVER, and hit rate are observed
% for the performance.

clear all; close all; clc;

load Golden_2009_2012_YearlyGroup_ClassGroup_TKCDfor_TKCDobs.mat

%% Data selection
yr = 2012;
KD_selYr = eval(strcat('KD_',num2str(yr)));
refKD = KD_selYr;

TCCDfor_selYr = eval(strcat('TCCDfor_',num2str(yr)));
TCCDfor = TCCDfor_selYr;

numDysYr = length(KD_selYr(:,1));

%% Estimating KD
% Relationship between KD and TCCDobs:
%  $KD_{hat\_trans} = -0.2825 * xdata - 0.1979$ ;
%  $KD_{hat} = (\lambda * KD_{hat\_trans} + 1)^{(1/\lambda)}$ ;

lambda = 2.0906;

```

```

TKCDfor = TCCDfor;
u_TKCDfor = 0.2;
u_a = 0.002908; u_b = 0.001327;
a = -0.2825; b = -0.1979;
sigma = 0.0565;

estKD = zeros(numDysYr,1);
estKD_up = zeros(numDysYr,1);
estKD_lo = zeros(numDysYr,1);

for i = 1:numDysYr
    % estimation
    if (-0.2825*TKCDfor(i)-0.1979)*lambda > -1
        estKD(i) = (lambda*(-0.2825*TKCDfor(i)-0.1979)+1)^(1/lambda);
    else
        estKD(i) = 0;
    end

    % upper
    if (-0.2825*TKCDfor(i)-0.1979+1.96*sigma)*lambda > -1
        estKD_up(i) = (lambda*(-0.2825*TKCDfor(i)-
0.1979+1.96*sigma)+1)^(1/lambda);
    else
        estKD_up(i) = 0;
    end

    % lower
    if (-0.2825*TKCDfor(i)-0.1979-1.96*sigma)*lambda > -1
        estKD_lo(i) = (lambda*(-0.2825*TKCDfor(i)-0.1979-
1.96*sigma)+1)^(1/lambda);
    else
        estKD_lo(i) = 0;
    end
end

for i = 1:numDysYr
    if TKCDfor(i) <= 0.4106
        estKD(i) = estKD(i) + 0.0559;
        estKD_up(i) = estKD_up(i) + 0.0559;
        estKD_lo(i) = estKD_lo(i) + 0.0559;
    end

    if estKD_up(i) > 1
        estKD_up(i) = 1;
    end
end

%% Classification by high, mid, and low KD level
numDys_hig = 0; numDys_mid = 0; numDys_low = 0;
idxDys_hig = zeros(numDysYr,1);
idxDys_mid = zeros(numDysYr,1);
idxDys_low = zeros(numDysYr,1);

for i = 1:numDysYr
    if refKD(i) >= 0.6

```

```

        numDys_hig = numDys_hig + 1;
        idxDys_hig(numDys_hig) = i;
    elseif refKD(i) < 0.3
        numDys_low = numDys_low + 1;
        idxDys_low(numDys_low) = i;
    else
        numDys_mid = numDys_mid + 1;
        idxDys_mid(numDys_mid) = i;
    end
end

idxDys_hig = idxDys_hig(1:numDys_hig);
idxDys_mid = idxDys_mid(1:numDys_mid);
idxDys_low = idxDys_low(1:numDys_low);

refKD_hig = refKD; refKD_hig = refKD_hig(idxDys_hig(:,1));
refKD_mid = refKD; refKD_mid = refKD_mid(idxDys_mid(:,1));
refKD_low = refKD; refKD_low = refKD_low(idxDys_low(:,1));

estKD_hig = estKD; estKD_hig = estKD_hig(idxDys_hig(:,1));
estKD_mid = estKD; estKD_mid = estKD_mid(idxDys_mid(:,1));
estKD_low = estKD; estKD_low = estKD_low(idxDys_low(:,1));

%% Result analysis for KD
idxInOut = zeros(numDysYr,3);
idxInOut(idxDys_hig(:,1),1) = 1;
idxInOut(idxDys_mid(:,1),1) = 2;
idxInOut(idxDys_low(:,1),1) = 3;

for i = 1:numDysYr
    if refKD(i) <= estKD_up(i) && refKD(i) >= estKD_lo(i)
        idxInOut(i,2) = 1;
    elseif refKD(i) > estKD_up(i)
        idxInOut(i,2) = 0;
        idxInOut(i,3) = refKD(i) - estKD_up(i);
    else
        idxInOut(i,2) = 0;
        idxInOut(i,3) = refKD(i) - estKD_lo(i);
    end
end

numIn = length(idxInOut(idxInOut(:,2)==1,2));

idxInOut_hig = idxInOut(idxInOut(:,1)==1,:);
numIn_hig = length(idxInOut_hig(idxInOut_hig(:,2)==1,2));

idxInOut_mid = idxInOut(idxInOut(:,1)==2,:);
numIn_mid = length(idxInOut_mid(idxInOut_mid(:,2)==1,2));

idxInOut_low = idxInOut(idxInOut(:,1)==3,:);
numIn_low = length(idxInOut_low(idxInOut_low(:,2)==1,2));

if sum(idxInOut(idxInOut(:,2)==0,3)) == 0
    MAOutDist = 0;
else

```

```

        MAOutDist = mean(abs(idxInOut(idxInOut(:,2)==0,3)));
        MBOutDist = mean(idxInOut(idxInOut(:,2)==0,3));
end

if sum(idxInOut_hig(idxInOut_hig(:,2)==0,3)) == 0
    MAOutDist_hig = 0;
    MBOutDist_hig = 0;
else
    MAOutDist_hig = mean(abs(idxInOut_hig(idxInOut_hig(:,2)==0,3)));
    MBOutDist_hig = mean(idxInOut_hig(idxInOut_hig(:,2)==0,3));
end

if sum(idxInOut_mid(idxInOut_mid(:,2)==0,3)) == 0
    MAOutDist_mid = 0;
    MBOutDist_mid = 0;
else
    MAOutDist_mid = mean(abs(idxInOut_mid(idxInOut_mid(:,2)==0,3)));
    MBOutDist_mid = mean(idxInOut_mid(idxInOut_mid(:,2)==0,3));
end

if sum(idxInOut_low(idxInOut_low(:,2)==0,3)) == 0
    MAOutDist_low = 0;
    MBOutDist_low = 0;
else
    MAOutDist_low = mean(abs(idxInOut_low(idxInOut_low(:,2)==0,3)));
    MBOutDist_low = mean(idxInOut_low(idxInOut_low(:,2)==0,3));
end

perIn = numIn/numDysYr * 100;
perIn_hig = numIn_hig/numDys_hig * 100;
perIn_mid = numIn_mid/numDys_mid * 100;
perIn_low = numIn_low/numDys_low * 100;

%% RMSE and MBE for KD
RMSE_KD = sqrt(mean((refKD - estKD).^2));
MBE_KD = mean(estKD - refKD);
MAE_KD = mean(abs(refKD - estKD));

RMSE_KD_hig = sqrt(mean((refKD_hig - estKD_hig).^2));
MBE_KD_hig = mean(refKD_hig - estKD_hig);
MAE_KD_hig = mean(abs(refKD_hig - estKD_hig));

RMSE_KD_mid = sqrt(mean((refKD_mid - estKD_mid).^2));
MBE_KD_mid = mean(refKD_mid - estKD_mid);
MAE_KD_mid = mean(abs(refKD_mid - estKD_mid));

RMSE_KD_low = sqrt(mean((refKD_low - estKD_low).^2));
MBE_KD_low = mean(refKD_low - estKD_low);
MAE_KD_low = mean(abs(refKD_low - estKD_low));

rRMSE_KD = sqrt(mean((refKD - estKD).^2)/mean(refKD.^2));
rRMSE_KD_hig = sqrt(mean((refKD_hig - estKD_hig).^2)/mean(refKD_hig.^2));
rRMSE_KD_mid = sqrt(mean((refKD_mid - estKD_mid).^2)/mean(refKD_mid.^2));
rRMSE_KD_low = sqrt(mean((refKD_low - estKD_low).^2)/mean(refKD_low.^2));

```

```

rMBE_KD = mean(estKD-refKD)/mean(refKD);
rMBE_KD_hig = mean(refKD_hig - estKD_hig)/mean(refKD_hig);
rMBE_KD_mid = mean(refKD_mid - estKD_mid)/mean(refKD_mid);
rMBE_KD_low = mean(refKD_low - estKD_low)/mean(refKD_low);

%% RMSE and MBE for TKCD
TCCDobs_selYr = eval(strcat('TCCDobs_',num2str(yr)));
refTKCD = TCCDobs_selYr;
estTKCD = TKCDfor;

refTKCD_hig = refTKCD; refTKCD_hig = refTKCD_hig(idxDys_hig(:,1));
refTKCD_mid = refTKCD; refTKCD_mid = refTKCD_mid(idxDys_mid(:,1));
refTKCD_low = refTKCD; refTKCD_low = refTKCD_low(idxDys_low(:,1));

estTKCD_hig = estTKCD; estTKCD_hig = estTKCD_hig(idxDys_hig(:,1));
estTKCD_mid = estTKCD; estTKCD_mid = estTKCD_mid(idxDys_mid(:,1));
estTKCD_low = estTKCD; estTKCD_low = estTKCD_low(idxDys_low(:,1));

RMSE_TKCD = sqrt(mean((refTKCD - estTKCD).^2));
RMSE_TKCD_hig = sqrt(mean((refTKCD_hig - estTKCD_hig).^2));
RMSE_TKCD_mid = sqrt(mean((refTKCD_mid - estTKCD_mid).^2));
RMSE_TKCD_low = sqrt(mean((refTKCD_low - estTKCD_low).^2));

rRMSE_TKCD = sqrt(mean((refTKCD - estTKCD).^2)/mean(refTKCD.^2));
rRMSE_TKCD_hig = sqrt(mean((refTKCD_hig - estTKCD_hig).^2)/mean(refTKCD_hig.^2));
rRMSE_TKCD_mid = sqrt(mean((refTKCD_mid - estTKCD_mid).^2)/mean(refTKCD_mid.^2));
rRMSE_TKCD_low = sqrt(mean((refTKCD_low - estTKCD_low).^2)/mean(refTKCD_low.^2));

MBE_TKCD = mean(estTKCD - refTKCD);
MBE_TKCD_hig = mean(refTKCD_hig - estTKCD_hig);
MBE_TKCD_mid = mean(refTKCD_mid - estTKCD_mid);
MBE_TKCD_low = mean(refTKCD_low - estTKCD_low);

rMBE_TKCD = mean(estTKCD-refTKCD)/mean(refTKCD);
rMBE_TKCD_hig = mean(refTKCD_hig - estTKCD_hig)/mean(refTKCD_hig);
rMBE_TKCD_mid = mean(refTKCD_mid - estTKCD_mid)/mean(refTKCD_mid);
rMBE_TKCD_low = mean(refTKCD_low - estTKCD_low)/mean(refTKCD_low);

%% KSI and OVER calculation
%%% Interpolation of 100 points
data_set1 = KD_2012;
data_set2 = estKD;

[f_set1,x_set1] = ecdf(data_set1);
[f_set2,x_set2] = ecdf(data_set2);

x_min_set1 = min(x_set1);
x_min_set2 = min(x_set2);

x_max_set1 = max(x_set1);
x_max_set2 = max(x_set2);

```

```

if x_min_set1 < x_min_set2
    x_min = x_min_set1;
else
    x_min = x_min_set2;
end
if x_max_set1 > x_max_set2
    x_max = x_max_set1;
else
    x_max = x_max_set2;
end

x_sam = (x_max - x_min) / 100;

xi = x_min:x_sam:x_max-x_sam;

f_inpl_set1 = zeros(100,1);
f_inpl_set2 = zeros(100,1);

for i = 1:100
    x_ref = xi(i);
    % set1
    for j = 2:length(x_set1)
        if x_ref < x_set1(j-1)
            f_inpl_set1(i) = 0;
            break;
        elseif x_ref >= x_set1(j-1) && x_ref < x_set1(j)
            f_inpl_set1(i) = f_set1(j-1);
            break;
        else
            f_inpl_set1(i) = 1;
        end
    end
    % set2
    for j = 2:length(x_set2)
        if x_ref < x_set2(j-1)
            f_inpl_set2(i) = 0;
            break;
        elseif x_ref >= x_set2(j-1) && x_ref < x_set2(j)
            f_inpl_set2(i) = f_set2(j-1);
            break;
        else
            f_inpl_set2(i) = 1;
        end
    end
end

%%% KSI calculation
%vc = 1.36/sqrt(length(data_set2)); % 95% CI
vc = 1.63/sqrt(length(data_set2)); % 99% CI

a_cri = vc*(x_max - x_min);
KSI = 100*sum(abs(f_inpl_set1-f_inpl_set2))*x_sam/a_cri;

%%% OVER calculation

```

```

extPnts = zeros(100,1);
overCnt = 0;
for i = 1:length(f_inpl_set1)

    if abs(f_inpl_set1(i)-f_inpl_set2(i))>vc
        overCnt = overCnt + 1;
        extPnts(overCnt) = abs(f_inpl_set1(i)-f_inpl_set2(i));
    end

end

extPnts = extPnts(1:overCnt);
OVER = 100*(sum(extPnts-vc))*x_sam/a_cri;

%% Plotting the hit rate result
f1 = figure(1);
lnWidth = 1.5;
mkSize = 7;
xaxis = 1:31;
lowDist = estKD(xaxis)-estKD_lo(xaxis);
uppDist = estKD_up(xaxis)-estKD(xaxis);
subplot(211)
errorbar(xaxis,estKD(xaxis),lowDist,uppDist,'ks',...
    'LineWidth',lnWidth,'MarkerFaceColor','k','MarkerSize',mkSize)
hold on
plot(xaxis,refKD(xaxis),'ro','MarkerFaceColor','r','MarkerSize',mkSize)
hold off
xlim([0 32]);ylim([0 1.3])
xlabel('Days'); ylabel('K_D');title('(e) Method 5')
legend('Forecast','Observation')
%print(f1,'-dtiff','-r500','SimulationResult_m5');

```

A.4. M-files for Chapter 5

A.4.1. 3-D synoptic view of K_D , POP- K_D , and standard deviation/kurtosis (Section 5.3.2)

● Synoptic_STD_kurtosis_11Yrs.m

```
% 3-D Synoptic View of KD, POP-KD, and STD or Kurtosis for 11 Years
% This script observes and plots the 3-D synoptic view of KD, POP-KD, and
% standard deviation/kurtosis at SRRL BMS for 11 years (2001 - 2011).

clear all; close all; clc;

%% Combining dataset for 11 years
begYr = 2001;
endYr = 2011;
KDYrs = comDatasetYrs('KD',begYr,endYr);
POPKDYrs = comDatasetYrs('POPKD',begYr,endYr);
stdQDYrs = comDatasetYrs('stdQD',begYr,endYr);
kurtoYrs = comDatasetYrs('kurto',begYr,endYr);

%% Plotting
fSize = 15;

% Standard deviation
f1 = figure(1);
ti1 = 0.04:.04:1;
ti2 = 0.42:.02:1;
[XI,YI] = meshgrid(ti1,ti2);
ZI = griddata(KDYrs,POPKDYrs,stdQDYrs,XI,YI);
bar3(ZI);
set(gca,'XTickLabel',[0.2 0.4 0.6 0.8 1.0])
set(gca,'YTickLabel',[0.40 0.50 0.60 0.70 0.80 0.90 1.0 1.2])
h_xlabel = xlabel('K_D');
h_ylabel = ylabel('POP-K_D');
h_zlabel = zlabel('\sigma');
set(h_xlabel,'FontSize',fSize);
set(h_ylabel,'FontSize',fSize);
set(h_zlabel,'FontSize',fSize);
%print(f1,'-dtiff','-r500','3D_STD');

% Kurtosis
f2 = figure(2);
ti1 = 0.04:.04:1;
ti2 = 0.42:.02:1;
[XI,YI] = meshgrid(ti1,ti2);
ZI = griddata(KDYrs,POPKDYrs,kurtoYrs,XI,1.4-YI);
bar3(ZI);
set(gca,'XTickLabel',[0.2 0.4 0.6 0.8 1.0])
set(gca,'YTickLabel',[1.00 0.90 0.80 0.70 0.60 0.50 0.40 0.30])
zlim([0 100])
h_xlabel = xlabel('K_D');
h_ylabel = ylabel('POP-K_D');
```



```

h_zlabel = zlabel('Kurtosis');
set(h_xlabel,'FontSize',fSize);
set(h_ylabel,'FontSize',fSize);
set(h_zlabel,'FontSize',fSize);
%print(f2,'-dtiff','-r500','3D_Kurto');

```

● comDatasetYrs.m

```

%% Combining Datasets for Selected Years
% This function combines yearly arrays of the selected data type, such as
% KD, POPKD, POPTKCDobs, POPTKCDfor, and others, for the selected years.

function comDataYrs = comDatasetYrs(dataTypStr,begYr,endYr)

%% Input
% dataTypStr: string of data type (KD,POPKD,POPTKCDobs,POPTKCDfor, and etc)
% dataLocStr: string of data location (Golden, UNLV, ORNL)
% begYr: beginning year
% endYr: ending year

%% Output
% comDataYrs: combined dataset for selected years

%% Load dataset
load Golden_2001_2012_YearlyGroup_KD_POPKD_Class_stdQD_kurto.mat
load Golden_2009_2012_YearlyGroup_POPTKCDfor_POPTKCDobs.mat

%% Selected data testing
if
strcmp(dataTypStr,'KD')||strcmp(dataTypStr,'POPKD')||strcmp(dataTypStr,'Class
')...
    ||strcmp(dataTypStr,'stdQD')||strcmp(dataTypStr,'kurto')
    dataStr = strcat(dataTypStr,'_Golden_');
elseif strcmp(dataTypStr,'POPTKCDfor')||strcmp(dataTypStr,'POPTKCDobs')...
    ||strcmp(dataTypStr,'POPTKCDhrobs')
    dataStr = strcat(dataTypStr,'_');
else
    disp('Selected data type does not exist in the loaded dataset!');
end

%% Combining

selYr = endYr - begYr + 1; % selected years

comDataYrs = zeros(selYr*365,1);

preDataPt = 0;
totLenDataYrs = 0;
for yr = 1:selYr
    refYr = begYr + (yr - 1);

    dataYr = eval(strcat(dataStr,num2str(refYr)));

```

```

lenDataYr = length(dataYr);

begPt = preDataPt + 1;
endPt = preDataPt + lenDataYr;

comDataYrs(begPt:endPt) = dataYr;

preDataPt = preDataPt + lenDataYr;
totLenDataYrs = totLenDataYrs + lenDataYr;
end

comDataYrs = comDataYrs(1:totLenDataYrs);

```

A.4.2. Ramp rate of I_{GHI} for 11 years (Section 5.3.3)

● RampRateGHI.m

```

%% Ramp Rate of GHI for 11 Years
% This script observes and plots the ramp rate of GHI at SRRL BMS for 11
% years (2001 - 2011) at different seasons and at different KD levels.

clear all; close all; clc;

load Golden_2001_2011_YearlyGroup_MonthlyGroup_GHI.mat
load Golden_2001_2012_YearlyGroup_idx_perK_proS.mat

%% Total 11-Year
range = 1500;
thre = 1;
distGHI = zeros(1,range/thre*2+1);

begYr = 2001;
endYr = 2011;

numDys = 0;
for yy = 1:endYr-begYr+1

    selYr = begYr - 1 + yy;

    dataset = eval(strcat('GHI_Golden_',num2str(selYr)));
    [rr cc] = size(dataset);

    endRow = 1;
    for i = 1:rr-1
        begRow = endRow + 1;
        if dataset(i,3)~=dataset(i+1,3)
            numDys = numDys + 1;
            endRow = i;
        end
    end
end

```

```

        irrDyTemp = dataset(begRow:endRow,6);
        distTemp = countRampRate(irrDyTemp,range,thre);
        distGHI = distGHI + distTemp;
    elseif i == rr-1
        numDys = numDys + 1;
        endRow = rr;
        irrDyTemp = dataset(begRow:endRow,6);
        distTemp = countRampRate(irrDyTemp,range,thre);
        distGHI = distGHI + distTemp;
    end
end
end

%% Summer
distGHI_Sum = zeros(1,range/thre*2+1);
numDys_Sum = 0;
mnSet = [5 6 7 8];
for mn = 1:length(mnSet)

    if mnSet(mn) < 10
        mnStr = strcat('0',num2str(mnSet(mn)));
    else
        mnStr = num2str(mnSet(mn));
    end
    dataset = eval(strcat('GHI_Golden_',mnStr,'_2001_2011'));
    [rr cc] = size(dataset);

    endRow = 1;
    for i = 1:rr-1
        begRow = endRow + 1;
        if dataset(i,3)~=dataset(i+1,3)
            numDys_Sum = numDys_Sum + 1;
            endRow = i;
            irrDyTemp = dataset(begRow:endRow,6);
            distTemp = countRampRate(irrDyTemp,range,thre);
            distGHI_Sum = distGHI_Sum + distTemp;
        elseif i == rr-1
            numDys_Sum = numDys_Sum + 1;
            endRow = rr;
            irrDyTemp = dataset(begRow:endRow,6);
            distTemp = countRampRate(irrDyTemp,range,thre);
            distGHI_Sum = distGHI_Sum + distTemp;
        end
    end
end

%% Spring&Fall
distGHI_SnF = zeros(1,range/thre*2+1);
numDys_SnF = 0;
mnSet = [3 4 9 10];
for mn = 1:length(mnSet)

    if mnSet(mn) < 10
        mnStr = strcat('0',num2str(mnSet(mn)));
    else
        mnStr = num2str(mnSet(mn));
    end
end

```

```

end
dataset = eval(strcat('GHI_Golden_',mnStr,'_2001_2011'));
[rr cc] = size(dataset);

endRow = 1;
for i = 1:rr-1
    begRow = endRow + 1;
    if dataset(i,3)~=dataset(i+1,3)
        numDys_SnF = numDys_SnF + 1;
        endRow = i;
        irrDyTemp = dataset(begRow:endRow,6);
        distTemp = countRampRate(irrDyTemp,range,thre);
        distGHI_SnF = distGHI_SnF + distTemp;
    elseif i == rr-1
        numDys_SnF = numDys_SnF + 1;
        endRow = rr;
        irrDyTemp = dataset(begRow:endRow,6);
        distTemp = countRampRate(irrDyTemp,range,thre);
        distGHI_SnF = distGHI_SnF + distTemp;
    end
end
end

%% Winter
distGHI_Win = zeros(1,range/thre*2+1);
numDys_Win = 0;
mnSet = [1 2 11 12];
for mn = 1:length(mnSet)

    if mnSet(mn) < 10
        mnStr = strcat('0',num2str(mnSet(mn)));
    else
        mnStr = num2str(mnSet(mn));
    end
    dataset = eval(strcat('GHI_Golden_',mnStr,'_2001_2011'));
    [rr cc] = size(dataset);

    endRow = 1;
    for i = 1:rr-1
        begRow = endRow + 1;
        if dataset(i,3)~=dataset(i+1,3)
            numDys_Win = numDys_Win + 1;
            endRow = i;
            irrDyTemp = dataset(begRow:endRow,6);
            distTemp = countRampRate(irrDyTemp,range,thre);
            distGHI_Win = distGHI_Win + distTemp;
        elseif i == rr-1
            numDys_Win = numDys_Win + 1;
            endRow = rr;
            irrDyTemp = dataset(begRow:endRow,6);
            distTemp = countRampRate(irrDyTemp,range,thre);
            distGHI_Win = distGHI_Win + distTemp;
        end
    end
end
end

```

```

%% Different KD levels
distGHI_Hig = zeros(1,range/thre*2+1);
distGHI_Mid = zeros(1,range/thre*2+1);
distGHI_Low = zeros(1,range/thre*2+1);

numDys_Hig = 0; numDys_Mid = 0; numDys_Low = 0;

totNumSam_Hig = 0;
totNumSam_Mid = 0;
totNumSam_Low = 0;

for yy = 1:endYr-begYr+1

    selYr = begYr - 1 + yy;

    dataset = eval(strcat('GHI_Golden_',num2str(selYr)));
    refKD = eval(strcat('perK_Golden_',num2str(selYr)));

    [rr cc] = size(dataset);

    numDysPerYr = 0;
    endRow = 1;
    for i = 1:rr-1
        begRow = endRow + 1;
        if dataset(i,3)~=dataset(i+1,3)
            numDysPerYr = numDysPerYr + 1;
            endRow = i;
            irrDyTemp = dataset(begRow:endRow,6);
            distTemp = countRampRate(irrDyTemp,range,thre);

            if refKD(numDysPerYr)>=0.6
                numDys_Hig = numDys_Hig + 1;
                distGHI_Hig = distGHI_Hig + distTemp;
                totNumSam_Hig = totNumSam_Hig + length(irrDyTemp)-1;
            elseif refKD(numDysPerYr)< 0.3
                numDys_Low = numDys_Low + 1;
                distGHI_Low = distGHI_Low + distTemp;
                totNumSam_Low = totNumSam_Low + length(irrDyTemp)-1;
            else
                numDys_Mid = numDys_Mid + 1;
                distGHI_Mid = distGHI_Mid + distTemp;
                totNumSam_Mid = totNumSam_Mid + length(irrDyTemp)-1;
            end

        elseif i == rr-1
            numDys = numDys + 1;
            numDysPerYr = numDysPerYr + 1;
            endRow = rr;
            irrDyTemp = dataset(begRow:endRow,6);
            distTemp = countRampRate(irrDyTemp,range,thre);

            if refKD(numDysPerYr)>=0.6
                numDys_Hig = numDys_Hig + 1;
                distGHI_Hig = distGHI_Hig + distTemp;
                totNumSam_Hig = totNumSam_Hig + length(irrDyTemp)-1;

```

```

elseif refKD(numDysPerYr)< 0.3
    numDys_Low = numDys_Low + 1;
    distGHI_Low = distGHI_Low + distTemp;
    totNumSam_Low = totNumSam_Low + length(irrDyTemp)-1;
else
    numDys_Mid = numDys_Mid + 1;
    distGHI_Mid = distGHI_Mid + distTemp;
    totNumSam_Mid = totNumSam_Mid + length(irrDyTemp)-1;
end
end
end
end

%% Generation of CDFs
totNumSam = totNumSam_Hig + totNumSam_Mid + totNumSam_Low;

% Total 11-year
refDist = distGHI/totNumSam;
magCount = zeros(1,range+1);
magCount(1) = refDist(range+1);
magCDF = zeros(1,range+1);
magCDF(1) = refDist(range+1);
for i = 1:range
    idx = range + 1;
    magCount(i+1) = refDist(idx+i) + refDist(idx-i);
    magCDF(i+1) = sum(magCount(1:i+1));
end

% High KD level
refDist = distGHI_Hig/totNumSam_Hig;
magCount_Hig = zeros(1,range+1);
magCount_Hig(1) = refDist(range+1);
magCDF_Hig = zeros(1,range+1);
magCDF_Hig(1) = refDist(range+1);
for i = 1:range
    idx = range + 1;
    magCount_Hig(i+1) = refDist(idx+i) + refDist(idx-i);
    magCDF_Hig(i+1) = sum(magCount_Hig(1:i+1));
end

% Medium KD level
refDist = distGHI_Mid/totNumSam_Mid;
magCount_Mid = zeros(1,range+1);
magCount_Mid(1) = refDist(range+1);
magCDF_Mid = zeros(1,range+1);
magCDF_Mid(1) = refDist(range+1);
for i = 1:range
    idx = range + 1;
    magCount_Mid(i+1) = refDist(idx+i) + refDist(idx-i);
    magCDF_Mid(i+1) = sum(magCount_Mid(1:i+1));
end

% Low KD level
refDist = distGHI_Low/totNumSam_Low;
magCount_Low = zeros(1,range+1);

```

```

magCount_Low(1) = refDist(range+1);
magCDF_Low = zeros(1,range+1);
magCDF_Low(1) = refDist(range+1);
for i = 1:range
    idx = range + 1;
    magCount_Low(i+1) = refDist(idx+i) + refDist(idx-i);
    magCDF_Low(i+1) = sum(magCount_Low(1:i+1));
end

%% Plotting
% Entire CDFs
f1 = figure(1);
plot([0:range],magCDF,'r-','LineWidth',3)
hold on
plot([0:range],magCDF_Hig,'b-','LineWidth',2)
plot([0:range],magCDF_Mid,'b-.','LineWidth',2)
plot([0:range],magCDF_Low,'k-','LineWidth',2)
hold off
grid on
xlim([0 800]);ylim([0 1])
h_legend = legend('Overall','High K_D','Medium K_D','Low K_D');
set(h_legend,'Location','SouthEast','FontSize',15);
h_xlabel = xlabel('Magnitude of Ramp Rate [W/m^2 per minute]');
h_ylabel = ylabel('Cumulative Frequency');
set(h_xlabel,'FontSize',15);
set(h_ylabel,'FontSize',15);
title('(a)')
%print(f1,'-dtiff','-r500','RampRate_a_GHI_CDFs_New');

% Zoomed between 0 and 150W/m2
f2 = figure(2);
plot([0:range],magCDF,'r-','LineWidth',3)
hold on
plot([0:range],magCDF_Hig,'b-','LineWidth',2)
plot([0:range],magCDF_Mid,'b-.','LineWidth',2)
plot([0:range],magCDF_Low,'k-','LineWidth',2)
plot(92.61,0:0.002:0.9515,'k.','LineWidth',.5)
plot(0:1:92.16,0.9515,'k.','LineWidth',.5)
text(0.003,0.9515+0.005,'0.95','FontSize',10)
text(92.16+1,0.81,'92.16','FontSize',10)
hold off
grid on
xlim([0 150]);ylim([0.8 1])
h_legend = legend('Overall','High K_D','Medium K_D','Low K_D');
set(h_legend,'Location','SouthEast','FontSize',15);
h_xlabel = xlabel('Magnitude of Ramp Rate [W/m^2 per minute]');
h_ylabel = ylabel('Cumulative Frequency');
set(h_xlabel,'FontSize',15);
set(h_ylabel,'FontSize',15);
title('(b)')
%print(f2,'-dtiff','-r500','RampRate_b_GHI_CDFs_New');

```

● countRampRate.m

```

%% Counting and Allocating Differences of GHIs for Distribution
% This function counts and allocates the differences of neighboring GHI
% values for the distribution.

function dist = countRampRate(inset,range,thre)
%% Input
% inset: n-by-1 (or 1-by-n) dataset of GHI or ETI
% range: range for distribution
% thre: threshold

%% Output
% dist: distribution of the counted values

%% Difference between neighboring samples

nSam = length(inset);
dInset = zeros(nSam-1,1);

for i = 1:nSam-1
    dInset(i) = inset(i+1) - inset(i);
end

%% Quantization by threshold
dInsetQuan = round(dInset/thre)*thre;

%% Counting
dist = zeros(1,range/thre*2+1);
for i = 1:nSam-1
    if dInsetQuan(i) == 0
        idx = int16(range/thre) + 1;
    else
        idx = int16(range/thre) + 1 + int16(dInsetQuan(i)/thre);
    end

    dist(idx) = dist(idx) + 1;
end

```

A.4.3. Relationship between $POP-TKC_{D,NWS}$ and $POP-TKC_D$ for 3 years (Section 5.4)

● POPTKCDNWSnPOPTKCD_3Yrs.m

```

%% CDFs of POP-TKCD,NWS and POP-TKCD for 3 Years
% This script compares CDFs of original values of POP-TKCD,NWS and
% POP-TKCD and CDFs of normalized values of them at SRRL BMS for 3 years
% (2009 - 2011).

clear all; close all; clc;

```



```

%% Load data
load Golden_2009_2012_YearlyGroup_POPTKCDfor_POPTKCDobs.mat

%% Original POP-TKCD
POP_TKCHr = POPTKCDfor_0911;
POP_TKCMn = POPTKCDobs_0911;
[f_TKCHr x_TKCHr] = ecdf(POP_TKCHr);
[f_TKCMn x_TKCMn] = ecdf(POP_TKCMn);

%% Normalized POP-TKCD
POP_TKCHr_NBR = (POP_TKCHr - min(POP_TKCHr))/(max(POP_TKCHr)-min(POP_TKCHr));
POP_TKCMn_NBR = (POP_TKCMn - min(POP_TKCMn))/(max(POP_TKCMn)-min(POP_TKCMn));
[f_TKCHr_NBR x_TKCHr_NBR] = ecdf(POP_TKCHr_NBR);
[f_TKCMn_NBR x_TKCMn_NBR] = ecdf(POP_TKCMn_NBR);

%% Plotting CDFs
lw = 3;
fSize = 15;

% Original CDFs
f1 = figure(1);
subplot(211)
plot(x_TKCMn,f_TKCMn,'r-','LineWidth',lw)
hold on
plot(x_TKCHr,f_TKCHr,'b-.','LineWidth',lw)
hold off
grid on
h_l = legend('POP-TKC_D','POP-TKC_D,_N_W_S');
h_y = ylabel('Cumulative Frequency');
h_t = title('(a)');
set(h_l,'Location','NorthWest','FontSize',fSize);
set(h_y,'FontSize',fSize);
set(h_t,'FontSize',fSize);

% Normalized CDFs
subplot(212)
plot(x_TKCMn_NBR,f_TKCMn_NBR,'r-','LineWidth',lw)
hold on
plot(x_TKCHr_NBR,f_TKCHr_NBR,'b-.','LineWidth',lw)
hold off
grid on
h_l = legend('Normalized POP-TKC_D','Normalized POP-TKC_D,_N_W_S');
h_y = ylabel('Cumulative Frequency');
h_t = title('(b)');
set(h_l,'Location','NorthWest','FontSize',fSize);
set(h_y,'FontSize',fSize);
set(h_t,'FontSize',fSize);
%print(f1,'-dtiff','-r500','CDF_POPTKCD_Original_NBR');

```

● VerificationNormalizationByRange.m

```

%% Forecasting Uncertainty Boundary (2012)

clear all; close all; clc;

load Golden_2012_YearlyGroup_KD_POPKD_TKCD_POPTKCD.mat
load Golden_2009_2012_YearlyGroup_POPTKCDfor_POPTKCDobs.mat

KDobs = KDobs_2012;
POPKDobs = POPKDobs_2012;
numDysYr = length(KDobs);
%% Forecasted and observed KD and POPKD
POPTKCD_NWS = POPTKCDfor_2012;
POPTKCD_for = (1-0.6737)*(POPTKCD_NWS-0.1250)/(1-0.1250)+0.6737;
POPKDfor = 1.694*POPTKCD_for-0.7471;

POPKDfor_up = 1.694*POPTKCD_for-0.5756;
POPKDfor_up(POPKDfor_up>1) = 1;
POPKDfor_lo = 1.694*POPTKCD_for-0.9186;

%% Classification by high, mid, and low radiation
numDys_hig = 0; numDys_mid = 0; numDys_low = 0;
idxDys_hig = zeros(numDysYr,1);
idxDys_mid = zeros(numDysYr,1);
idxDys_low = zeros(numDysYr,1);

for i = 1:numDysYr
    if KDobs(i) >= 0.6
        numDys_hig = numDys_hig + 1;
        idxDys_hig(numDys_hig) = i;
    elseif KDobs(i) < 0.3
        numDys_low = numDys_low + 1;
        idxDys_low(numDys_low) = i;
    else
        numDys_mid = numDys_mid + 1;
        idxDys_mid(numDys_mid) = i;
    end
end

idxDys_hig = idxDys_hig(1:numDys_hig);
idxDys_mid = idxDys_mid(1:numDys_mid);
idxDys_low = idxDys_low(1:numDys_low);

KDobs_hig = KDobs; KDobs_hig = KDobs_hig(idxDys_hig(:,1));
KDobs_mid = KDobs; KDobs_mid = KDobs_mid(idxDys_mid(:,1));
KDobs_low = KDobs; KDobs_low = KDobs_low(idxDys_low(:,1));

%% Counting in or out of the boundary
numIn = 0;
numIn_hig = 0;
numIn_mid = 0;
numIn_low = 0;

for i = 1:numDysYr
    if POPKDobs(i)>=POPKDfor_lo(i) && POPKDobs(i)<=POPKDfor_up(i)
        numIn = numIn + 1;
    end
end

```

```

        if KDobs(i) >= 0.6
            numIn_hig = numIn_hig + 1;
        elseif KDobs(i) < 0.3
            numIn_low = numIn_low + 1;
        else
            numIn_mid = numIn_mid + 1;
        end
    end
end

%% Plotting analysis results
f1 = figure(1);
lnWidth = 1.5;
mkSize = 7;
xaxis = 1:31;
lowDist = POPKDfor(xaxis)-POPKDfor_lo(xaxis);
uppDist = POPKDfor_up(xaxis)-POPKDfor(xaxis);
subplot(211)
errorbar(xaxis,POPKDfor(xaxis),lowDist,uppDist,'ks',...
    'LineWidth',lnWidth,'MarkerFaceColor','k','MarkerSize',mkSize)
hold on
plot(xaxis,POPKDobs(xaxis),'ro','MarkerFaceColor','r','MarkerSize',mkSize)
hold off
xlim([0 32]);ylim([0 1.1])

h_x = xlabel('Days');
h_y = ylabel('POP-K_D');
h_l = legend('Location','SouthEast','Estimation','Observation');
set(h_l,'FontSize',15);
set(h_x,'FontSize',15);
set(h_y,'FontSize',15);

print(f1,'-dtiff','-r500','SimulationResult_Quality');

```

A.4.4. Relationship between POP-TKCD and POP-KD for 3 years (Section 5.5)

● POPTKCDnPOPKD_3Yrs.m

```

%% Plotting POP-TKCD and POP-KD for 3 Years
% This script plots the relationship between daily observed POP of total
% sky cover (POP-TKCD) and daily observed POP of solar irradiance (POP-KD)
% at SRRL BMS for 3 years (2009 - 2011).
% This script also shows the fitted line derived from this relationship.

clear all; close all; clc;

load Golden_2001_2012_YearlyGroup_idx_perK_proS.mat
load Golden_2006_2012_YearlyGroup_TKCD_OKCD_POPTKCD_POPOKCD.mat

%% Combining datasets
xx = 0:0.01:1;

```

```

POPTKC_3yr = zeros(3*365,1);
POPTKC_3yr(1:365) = POPTCC_Golden_2009;
POPTKC_3yr(365+1:365*2) = POPTCC_Golden_2010;
POPTKC_3yr(2*365+1:365*3) = POPTCC_Golden_2011;

POPKD_3yr = zeros(3*365,1);
POPKD_3yr(1:365) = proS_Golden_2009;
POPKD_3yr(365+1:365*2) = proS_Golden_2010;
POPKD_3yr(2*365+1:365*3) = proS_Golden_2011;

%% Plotting
lw = 3;
mkSize = 15;
fSize = 15;

f1 = figure(1);
plot(POPTKC_3yr,POPKD_3yr,'k.','MarkerSize',mkSize)
hold on
plot(xx,1.677*xx-0.7353,'r-','LineWidth',lw)
hold off
grid on
xlim([0.6 1]);ylim([0.4 1])
h_xlabel = xlabel('POP-TKC_D');
h_ylabel = ylabel('POP-K_D');
h_legend = legend('Data point','Fitted line');
set(h_legend,'Location','NorthWest','FontSize',fSize);
set(h_xlabel,'FontSize',fSize);
set(h_ylabel,'FontSize',fSize);
%print(f1,'-dtiff','-r500','POPKDPOPTKCD3Yrs');

```

● VerificationConversionLine.m

```

%% Verification of Conversion Line
% This script verifies the derived conversion line from POP-TKCD to POP-KD
% based on the comparison between the observed POP-KD and the estimated
% POP-KD converted from the observed POP-TKCD using the 2012 data.

clear all; close all; clc;

load Golden_2009_2012_YearlyGroup_POPTKCDfor_POPTKCDobs.mat
load Golden_2012_YearlyGroup_KD_POPKD_TKCD_POPTKCD.mat
%% Conversion from POP-TKCD to POP-KD
POPKD_obs = POPKDobs_2012;

POPTKCD_obs = POPTKCDobs_2012;
POPKD_for = 1.677*POPTKCD_obs-0.7353;
%% Classification by high, mid, and low KD level
refKD = KDobs_2012;
numDysYr = length(refKD);

numDys_hig = 0; numDys_mid = 0; numDys_low = 0;

```

```

idxDys_hig = zeros(numDysYr,1);
idxDys_mid = zeros(numDysYr,1);
idxDys_low = zeros(numDysYr,1);

for i = 1:numDysYr
    if refKD(i) >= 0.6
        numDys_hig = numDys_hig + 1;
        idxDys_hig(numDys_hig) = i;
    elseif refKD(i) < 0.3
        numDys_low = numDys_low + 1;
        idxDys_low(numDys_low) = i;
    else
        numDys_mid = numDys_mid + 1;
        idxDys_mid(numDys_mid) = i;
    end
end

idxDys_hig = idxDys_hig(1:numDys_hig);
idxDys_mid = idxDys_mid(1:numDys_mid);
idxDys_low = idxDys_low(1:numDys_low);

POPKD_obs_hig = POPKD_obs; POPKD_obs_hig = POPKD_obs_hig(idxDys_hig(:,1));
POPKD_obs_mid = POPKD_obs; POPKD_obs_mid = POPKD_obs_mid(idxDys_mid(:,1));
POPKD_obs_low = POPKD_obs; POPKD_obs_low = POPKD_obs_low(idxDys_low(:,1));

POPKD_for_hig = POPKD_for; POPKD_for_hig = POPKD_for_hig(idxDys_hig(:,1));
POPKD_for_mid = POPKD_for; POPKD_for_mid = POPKD_for_mid(idxDys_mid(:,1));
POPKD_for_low = POPKD_for; POPKD_for_low = POPKD_for_low(idxDys_low(:,1));

%% RMSE and MBE
RMSE_POPKD = sqrt(mean((POPKD_obs - POPKD_for).^2));
MBE_POPKD = mean(POPKD_for - POPKD_obs);
MAE_POPKD = mean(abs(POPKD_obs - POPKD_for));

RMSE_POPKD_hig = sqrt(mean((POPKD_obs_hig - POPKD_for_hig).^2));
MBE_POPKD_hig = mean(POPKD_obs_hig - POPKD_for_hig);
MAE_POPKD_hig = mean(abs(POPKD_obs_hig - POPKD_for_hig));

RMSE_POPKD_mid = sqrt(mean((POPKD_obs_mid - POPKD_for_mid).^2));
MBE_POPKD_mid = mean(POPKD_obs_mid - POPKD_for_mid);
MAE_POPKD_mid = mean(abs(POPKD_obs_mid - POPKD_for_mid));

RMSE_POPKD_low = sqrt(mean((POPKD_obs_low - POPKD_for_low).^2));
MBE_POPKD_low = mean(POPKD_obs_low - POPKD_for_low);
MAE_POPKD_low = mean(abs(POPKD_obs_low - POPKD_for_low));

rRMSE_POPKD = sqrt(mean((POPKD_obs - POPKD_for).^2)/mean(POPKD_obs.^2));
rRMSE_POPKD_hig = sqrt(mean((POPKD_obs_hig - POPKD_for_hig).^2)/mean(POPKD_obs_hig.^2));
rRMSE_POPKD_mid = sqrt(mean((POPKD_obs_mid - POPKD_for_mid).^2)/mean(POPKD_obs_mid.^2));
rRMSE_POPKD_low = sqrt(mean((POPKD_obs_low - POPKD_for_low).^2)/mean(POPKD_obs_low.^2));

rMBE_POPKD = mean(POPKD_for-POPKD_obs)/mean(POPKD_obs);

```

```

rMBE_POPKD_hig = mean(POP KD_obs_hig - POP KD_for_hig)/mean(POP KD_obs_hig);
rMBE_POPKD_mid = mean(POP KD_obs_mid - POP KD_for_mid)/mean(POP KD_obs_mid);
rMBE_POPKD_low = mean(POP KD_obs_low - POP KD_for_low)/mean(POP KD_obs_low);

```

A.4.5. Description of the proposed quality method (Section 5.6)

● CIBoundaries_QualityMethod.m

```

%% Conversion Line with 95% CIs for Quality Method
% This script plots the figures that the proposed conversion line with its
% 95% CI boundaries for the proposed quality estimation method.

clear all; close all; clc;

xx = 0:0.001:1;
yy = 1.694*xx-0.7471;

sigmaT = zeros(size(xx));
yy_up = zeros(size(xx));
yy_lo = zeros(size(xx));

for i = 1:length(xx)
    sigmaT(i) = sqrt((xx(i)*0.02041)^2+0.0185^2+(1.694*0.05)^2);
    yy_up(i) = yy(i) + 2*sigmaT(i);
    yy_lo(i) = yy(i) - 2*sigmaT(i);
end

obsPnt = 750;

%% Plotting
f1 = figure(1);
plot(xx,yy_up,'r--','LineWidth',1)
hold on
plot(xx,yy,'r-','LineWidth',2)
plot(xx,yy_lo,'r--','LineWidth',1)

% estimation
plot(xx(obsPnt),0:0.01:yy(obsPnt),'k.','LineWidth',.5)
plot(0:0.005:xx(obsPnt),yy(obsPnt),'k.','LineWidth',.5)
text(0.61,yy(obsPnt)+0.03,'POP-K_D,_f_o_r','FontSize',10)

% upper
plot(xx(obsPnt),0:0.01:yy_up(obsPnt),'k.','LineWidth',.5)
plot(0:0.005:xx(obsPnt),yy_up(obsPnt),'k.','LineWidth',.5)
text(0.61,yy_up(obsPnt)+0.03,'Upper boundary of POP-
K_D,_f_o_r','FontSize',10)

% lower
plot(xx(obsPnt),0:0.01:yy_lo(obsPnt),'k.','LineWidth',.5)
plot(0:0.005:xx(obsPnt),yy_lo(obsPnt),'k.','LineWidth',.5)

```

```

text(0.61,yy_lo(obsPnt)+0.03,'Lower boundary of POP-
K_D,_f_o_r','FontSize',10)

hold off
legend('95% CI (upper)','Conversion line',...
       '95% CI (lower)','Location','NorthWest')
xlabel('POP-TKC_D,_f_o_r'); ylabel('POP-K_D,_f_o_r')
ylim([0.001 1]);xlim([0.6 1])
title('95% Confidence Interval')

%print(f1,'-dtiff','-r500','UncertaintyBoundary');

```

A.4.6. Tests for the proposed quality method (Section 5.7)

● EstimationTest_QualityMethod_2012.m

```

%% Estimation Test of Quality Method Using the 2012 Data
% This script tests the estimation performance of the proposed quality
% method using the 2012 data. The RMSE, MBE, KSI, OVER, and hit rate are
% observed for the performance.

clear all; close all; clc;

load Golden_2009_2012_YearlyGroup_POPTKCDfor_POPTKCDobs.mat
load Golden_2012_YearlyGroup_KD_POPKD_TKCD_POPTKCD.mat

KDobs = KDobs_2012;
POPKDobs = POPKDobs_2012;
numDysYr = length(KDobs);

%% Estimated and observed KD and POPKD
POPTKCD_NWS = POPTKCDfor_2012;
POPTKCD_for = (1-0.6737)*(POPTKCD_NWS-0.1250)/(1-0.1250)+0.6737;
POPKDfor = 1.694*POPTKCD_for-0.7471;

POPKDfor_up = 1.694*POPTKCD_for-0.5756;
POPKDfor_up(POPKDfor_up>1) = 1;
POPKDfor_lo = 1.694*POPTKCD_for-0.9186;

%% RMES, MBE, and MAE
refKD = KDobs_2012;

numDys_hig = 0; numDys_mid = 0; numDys_low = 0;
idxDys_hig = zeros(numDysYr,1);
idxDys_mid = zeros(numDysYr,1);
idxDys_low = zeros(numDysYr,1);

for i = 1:numDysYr
    if refKD(i) >= 0.6
        numDys_hig = numDys_hig + 1;
        idxDys_hig(numDys_hig) = i;
    end
end

```

```

elseif refKD(i) < 0.3
    numDys_low = numDys_low + 1;
    idxDys_low(numDys_low) = i;
else
    numDys_mid = numDys_mid + 1;
    idxDys_mid(numDys_mid) = i;
end
end

idxDys_hig = idxDys_hig(1:numDys_hig);
idxDys_mid = idxDys_mid(1:numDys_mid);
idxDys_low = idxDys_low(1:numDys_low);

POPKDobs_hig = POPKDobs; POPKDobs_hig = POPKDobs_hig(idxDys_hig(:,1));
POPKDobs_mid = POPKDobs; POPKDobs_mid = POPKDobs_mid(idxDys_mid(:,1));
POPKDobs_low = POPKDobs; POPKDobs_low = POPKDobs_low(idxDys_low(:,1));

POPKDfor_hig = POPKDfor; POPKDfor_hig = POPKDfor_hig(idxDys_hig(:,1));
POPKDfor_mid = POPKDfor; POPKDfor_mid = POPKDfor_mid(idxDys_mid(:,1));
POPKDfor_low = POPKDfor; POPKDfor_low = POPKDfor_low(idxDys_low(:,1));

RMSE_POPKD = sqrt(mean((POPKDobs - POPKDfor).^2));
MBE_POPKD = mean(POPKDfor - POPKDobs);
MAE_POPKD = mean(abs(POPKDobs - POPKDfor));

RMSE_POPKD_hig = sqrt(mean((POPKDobs_hig - POPKDfor_hig).^2));
MBE_POPKD_hig = mean(POPKDobs_hig - POPKDfor_hig);
MAE_POPKD_hig = mean(abs(POPKDobs_hig - POPKDfor_hig));

RMSE_POPKD_mid = sqrt(mean((POPKDobs_mid - POPKDfor_mid).^2));
MBE_POPKD_mid = mean(POPKDobs_mid - POPKDfor_mid);
MAE_POPKD_mid = mean(abs(POPKDobs_mid - POPKDfor_mid));

RMSE_POPKD_low = sqrt(mean((POPKDobs_low - POPKDfor_low).^2));
MBE_POPKD_low = mean(POPKDobs_low - POPKDfor_low);
MAE_POPKD_low = mean(abs(POPKDobs_low - POPKDfor_low));

rRMSE_POPKD = sqrt(mean((POPKDobs - POPKDfor).^2)/mean(POPKDobs.^2));
rRMSE_POPKD_hig = sqrt(mean((POPKDobs_hig - POPKDfor_hig).^2)/mean(POPKDobs_hig.^2));
rRMSE_POPKD_mid = sqrt(mean((POPKDobs_mid - POPKDfor_mid).^2)/mean(POPKDobs_mid.^2));
rRMSE_POPKD_low = sqrt(mean((POPKDobs_low - POPKDfor_low).^2)/mean(POPKDobs_low.^2));

rMBE_POPKD = mean(POPKDfor-POPKDobs)/mean(POPKDobs);
rMBE_POPKD_hig = mean(POPKDobs_hig - POPKDfor_hig)/mean(POPKDobs_hig);
rMBE_POPKD_mid = mean(POPKDobs_mid - POPKDfor_mid)/mean(POPKDobs_mid);
rMBE_POPKD_low = mean(POPKDobs_low - POPKDfor_low)/mean(POPKDobs_low);

%% KSI and OVER calculation
%% Interpolation of 100 points
data_set1 = POPKDobs;
data_set2 = POPKDfor;

```



```

[f_set1,x_set1] = ecdf(data_set1);
[f_set2,x_set2] = ecdf(data_set2);

x_min_set1 = min(x_set1);
x_min_set2 = min(x_set2);

x_max_set1 = max(x_set1);
x_max_set2 = max(x_set2);

if x_min_set1 < x_min_set2
    x_min = x_min_set1;
else
    x_min = x_min_set2;
end
if x_max_set1 > x_max_set2
    x_max = x_max_set1;
else
    x_max = x_max_set2;
end

%x_min = min(x_set2); x_max = max(x_set2);

x_sam = (x_max - x_min) / 100;

xi = x_min:x_sam:x_max-x_sam;

f_inpl_set1 = zeros(100,1);
f_inpl_set2 = zeros(100,1);

for i = 1:100
    x_ref = xi(i);
    % set1
    for j = 2:length(x_set1)
        if x_ref < x_set1(j-1)
            f_inpl_set1(i) = 0;
            break;
        elseif x_ref >= x_set1(j-1) && x_ref < x_set1(j)
            f_inpl_set1(i) = f_set1(j-1);
            break;
        else
            f_inpl_set1(i) = 1;
        end
    end
    % set2
    for j = 2:length(x_set2)
        if x_ref < x_set2(j-1)
            f_inpl_set2(i) = 0;
            break;
        elseif x_ref >= x_set2(j-1) && x_ref < x_set2(j)
            f_inpl_set2(i) = f_set2(j-1);
            break;
        else
            f_inpl_set2(i) = 1;
        end
    end
end

```

```

end

%%% KSI calculation
%vc = 1.36/sqrt(length(data_set2)); % 95% CI
vc = 1.63/sqrt(length(data_set2)); % 99% CI

a_cri = vc*(x_max - x_min);
KSI = 100*sum(abs(f_inpl_set1-f_inpl_set2))*x_sam/a_cri;

%%% OVER calculation
extPnts = zeros(100,1);
overCnt = 0;
for i = 1:length(f_inpl_set1)

    if abs(f_inpl_set1(i)-f_inpl_set2(i))>vc
        overCnt = overCnt + 1;
        extPnts(overCnt) = abs(f_inpl_set1(i)-f_inpl_set2(i));
    end

end

extPnts = extPnts(1:overCnt);
OVER = 100*(sum(extPnts-vc))*x_sam/a_cri;

%% Hit rate observation
numIn = 0;
numIn_hig = 0;
numIn_mid = 0;
numIn_low = 0;

for i = 1:numDysYr
    if POPKDobs(i)>=POPKDfor_lo(i) && POPKDobs(i)<=POPKDfor_up(i)
        numIn = numIn + 1;
        if KDobs(i) >= 0.6
            numIn_hig = numIn_hig + 1;
        elseif KDobs(i) < 0.3
            numIn_low = numIn_low + 1;
        else
            numIn_mid = numIn_mid + 1;
        end
    end
end

end

%% Plotting
% CDF
[f_for, x_for] = ecdf(POPKDfor);
[f_obs, x_obs] = ecdf(POPKDobs);
f1 = figure(1);
plot(x_obs,f_obs,'k-.','LineWidth',1.5)
hold on
plot(x_for,f_for,'r-','LineWidth',1.5)
hold off
grid on
xlabel('POP-K_D,_f_o_r');ylabel('Cumulative Frequency (F)')
title('(a) CDFs')
legend('Observation','Forecast','Location','NorthWest')

```

```

%print(f1,'-dtiff','-r500','ForecastingKD2012_CDF');

% KSI plot
f2 = figure(2);
plot(xi,abs(f_inpl_set1-f_inpl_set2),'r-*)
hold on
plot(0:0.01:1-0.01,vc*ones(100,1),'k-.')
text(0.45,vc+0.005,'Threshold','FontSize',10)
hold off
grid on
xlim([0.4 0.9772]);ylim([0 0.2])
xlabel('POP-K_D,_f_o_r');ylabel('Difference')
legend('Forecasting Result','Location','NorthWest')
title('(b) Differences of CDFs')
%print(f2,'-dtiff','-r500','ForecastingKD2012_Dn');

% Plotting the hit rate
f3 = figure(3);
lnWidth = 1.5;
mkSize = 7;
xaxis = 1:31;
lowDist = POPKDfor(xaxis)-POPKDfor_lo(xaxis);
uppDist = POPKDfor_up(xaxis)-POPKDfor(xaxis);
subplot(211)
errorbar(xaxis,POPKDfor(xaxis),lowDist,uppDist,'ks',...
'LineWidth',lnWidth,'MarkerFaceColor','k','MarkerSize',mkSize)
hold on
plot(xaxis,POPKDobs(xaxis),'ro','MarkerFaceColor','r','MarkerSize',mkSize)
hold off
xlim([0 32]);ylim([0 1.1])

h_x = xlabel('Days');
h_y = ylabel('POP-K_D');
h_l = legend('Location','SouthEast','Estimation','Observation');
set(h_l,'FontSize',15);
set(h_x,'FontSize',15);
set(h_y,'FontSize',15);

%print(f3,'-dtiff','-r500','SimulationResult_Quality');

```

REFERENCE

- Aguiar, R.J., Collares-Pereira, M., Conde, J.P., 1988. Simple procedure for generating sequences of daily radiation values using a library of Markov transition matrices. *Solar Energy* 40, 269–279.
- Al-Alawi, S.M., Al-Hinai, H.A., 1998. An ANN-based approach for predicting global radiation in locations with no direct measurement instrumentation. *Renewable Energy* 14, 199–204.
- Alam, S., Kaushik, S.C., Garg, S.N., 2006. Computation of beam solar radiation at normal incidence using artificial neural network. *Renewable energy* 31, 1483–1491.
- Assuncao, H.F., Escobedo, J.F., Oliveira, A.P., 2007. A new algorithm to estimate sky condition based on 5 minutes-averaged values of clearness index and relative optical air mass. *Theoretical and Applied Climatology* 90, 235–248
- Badescu, V., 2008. Modeling solar radiation at the earth's surface: recent advances. Springer, London.
- Baharuddin, Lau, S.S.Y., Rahim, R., 2010. Daylight availability in Hong Kong: classification into three sky conditions. *Architectural Science Review* 53, 396–407.
- Bendt, P., Collares-Pereira, M., Rabl, A., 1981. The frequency distribution of daily insolation values. *Solar Energy* 27, 1–5.
- Box, G.E.P., Cox, D.R., 1964. An analysis of transformations. *Journal of the Royal Statistical Society Series B*, 211–252.
- Calbo, J., Gonzalez, J.A., Pages, D., 2001. A method for sky-condition classification from ground-based solar radiation measurements. *Applied Meteorology* 40, 2193–2199.

- Cazorla, A., Olmo, F.J., Alados-Arboledas, L., 2008. Development of a sky imager for cloud cover assessment. *JOSA A* 25, 29–39.
- Che, H.Z., Shi, G.Y., Zhang, X.Y., Zhao, J.Q., Li, Y., 2007. Analysis of sky conditions using 40 year records of solar radiation data in China. *Theoretical and Applied Climatology* 89, 83–94.
- Chow, W.C., Urquhart, B., Lave, M., Dominquez, A., Kleissl, J., Shields, J., Washom, B., 2011. Intra-hour forecasting with a total sky imager at the UC San Diego solar energy testbed. *Solar Energy* 85, 2881–2893.
- Corder, G.W., Foreman, D.I., 2009. *Nonparametric statistics for non-statisticians*. John Wiley & Sons, Hoboken.
- Crispim, E.M., Ferreira, P.M., Ruano, A.E., 2008. Prediction of the solar radiation evolution using computational intelligence techniques and cloudiness indices. *International Journal of Innovative Computing, Information and Control* 4, 1121–1133.
- Dorvlo, A.S., Jervase, J.A., Al-Lawati, A., 2002. Solar radiation estimation using artificial neural networks. *Applied Energy* 71, 307–319.
- Ellison, S.L.R., Williams, A., 2012. Eurachem/CITAC guide: quantifying uncertainty in analytical measurement. EURACHEM/CITAC, <<http://www.eurachem.org>>.
- Escrig, H., Batlles, F.J., Alonso, J., Baena, F.M., Bosch, J.L., Salbidegoitia, I.L., Burgaleta, J.I., 2013. Cloud detection, classification and motion estimation using geostationary satellite imagery for cloud cover forecast. *Energy* 55, 853–859.
- Espinar, B., Ramirez, L., Drews, A., Beyer, H.G., Zarzalejo, L.F., Polo, J., Martin, L., 2009. Analysis of difference comparison parameters applied to solar radiation data from satellite and German radiometric stations. *Solar Energy* 83, 118–125.

- Gautam, N.K., Kaushika, N.D., 2002. A model for the estimation of global solar radiation using fuzzy random variables. *Journal of Applied Meteorology* 41, 1267–1276.
- Gomez, V., Casanovas, A., 2003. Fuzzy modeling of solar irradiance on inclined surfaces. *Solar Energy* 75, 307–315.
- Hammer, A., Heinemann, D., Lorenz, E., Luckehe, B., 1999. Short-term forecasting of solar radiation: a statistical approach using satellite data. *Solar Energy* 67, 139–150.
- Hansen, J.W., 1999. Stochastic daily solar irradiance for biological modeling applications¹. *Agricultural and forest meteorology* 94, 53–63.
- Harrouni, S., Guessoum, A., Maafi, A., 2005. Classification of daily solar irradiation by fractional analysis of 10-min-means of solar irradiance. *Theoretical and Applied Climatology* 80, 27–36.
- Harvey, A.C., 1990. *Forecasting, structural time series models and the Kalman filter*. Cambridge University Press, Cambridge.
- Heinemann, D., Lorenz, E., Girodo, M., 2006. *Forecasting of solar radiation. Solar energy resource management for electricity generation from local level to global scale*. Nova Science Publishers, New York.
- Ibanez, M., Beckman, W.A., Kein, S.A., 2002. Frequency distributions for hourly and daily clearness indices. *J. Solar Energy Eng.* 124, 28–33.
- Ibanez, M., Rosell, J., Beckman, W., 2003. A bi-variable probability density function for the daily clearness index. *Solar Energy* 75, 73–80.
- Iqbal, M., 1983. *An introduction to solar radiation*. Academic Press, Orlando.
- Jurado, M., Caridad, J.M., Ruiz, V., 1995. Statistical distribution of the clearness index with radiation data integrated over five minute intervals. *Solar Energy* 55, 469–473.

- Kang, B.O., Tam, K.-S., 2013. A new characterization and classification method for daily sky conditions based on ground-based solar irradiance measurement data. *Solar Energy* 94, 102–118.
- Kasten, F., Czeplak, G., 1979. Solar and terrestrial radiation dependent on the amount and type of cloud. *Solar Energy* 24, 177–189.
- Kawasaki, N., Oozeki, T., Otani, K., Kurokawa, K., 2006. An evaluation method of the fluctuation characteristics of photovoltaic systems by using frequency analysis. *Solar Energy Materials and Solar Cells* 90, 3356–3363.
- Kazantzidis, A., Tzoumanikas, P., Bais, A.F., Fotopoulos, S., Economou, G., 2012. Cloud detection and classification with the use of whole-sky ground-based images. *Atmospheric Research* 113, 80–88.
- Kirby, B., Milligan, M., 2005. A method and case study for estimating the ramping capability of a control area or balancing authority and implications for moderate or high wind penetration. In: *Proc. of the WINDPOWER 2005 Conference*. NREL/CP-500-38153.
- Kitamura, A., 1999. Demonstration test results for grid interconnected photovoltaic power systems. Report IEA-PVPS T5-02: 1999.
- Kleissl, J., 2010. Current state of the art in solar forecasting, Appendix, California renewable energy forecasting, resource data, and mapping. CIEE, BOA-99-248-R.
- Kratzenberg, M.G., Colle, S., Beyer, H.G., 2008. Solar radiation prediction based on the combination of a numerical weather prediction model and a time series prediction model. In: *Proc. of the EUROSUN 2008*, Lisbon, Portugal.
- Liu, B.Y.H., Jordan, R.C., 1960. The interrelationship and characteristic distribution of direct, diffuse and total solar radiation. *Solar Energy* 4, 1–19.

- Long, C.N., Sabburg, J.M., Calbo, J., Pages, D., 2006. Retrieving cloud characteristics from ground-based daytime color all-sky images. *Journal of Atmospheric and Oceanic Technology* 23, 633–652.
- Maafi, S., Harrouni, S., 2003. Preliminary results of the fractal classification of daily solar irradiances. *Solar Energy* 75, 53–61.
- Marquez, R., Coimbra, C.F.M., 2011. Forecasting of global and direct solar irradiance using stochastic learning methods, ground experiments and the NWS database. *Solar Energy* 85, 746–756.
- Marquez, R., Gueorguiev, V.G., Coimbra, C.F.M., 2013. Forecasting of global horizontal irradiance using sky cover indices. *Journal of Solar Energy Engineering* 135, 5.
- Marquez, R., Coimbra, C.F.M., 2013. Intra-hour DNI forecasting based on cloud tracking image analysis. *Solar Energy* 91, 327 – 336.
- Mathiesen, P., Kleissl, J., 2011. Evaluation of numerical weather prediction for intra-day solar forecasting in the continental United States. *Solar Energy* 85, 967–977.
- Mellit, A., Benghane, M., Kalogirou, S.A., 2006. An adaptive wavelet-network model for forecasting daily total solar-radiation. *Applied Energy* 83, 705–722.
- Mellit, A., HadjArab, A., Khorissi, N., Salhi, H., 2007. An ANFIS-based forecasting for solar radiation data from sunshine duration and ambient temperature. In: *Proc. of 2007 PES General Meeting*, Tampa, FL.
- Mellit, A., Kalogirou, S.A., 2008. Artificial intelligence techniques for photovoltaic applications: A review. *Progress in Energy and Combustion Science*, 34, 574–632.
- Meyn, S.P., Tweedie, R.L., Glynn, P.W., 2009. *Markov chains and stochastic stability*. Cambridge University Press, Cambridge.

- MIDC, the Measurement and Instrumentation Data Center, National Renewable Energy Laboratory, 2012. Washington, DC. <www.nrel.gov/midc/>.
- Mohandes, M., Rehman, S., Halawani, T.O., 1998. Estimation of global solar radiation using artificial neural networks. *Renewable Energy*, 14, 179–184.
- Mora-Lopez, L.L., Sidrach-de-Cardona, M., 1998. Multiplicative ARMA models to generate hourly series of global irradiation. *Solar Energy* 63, 283–291.
- NCDC, the National Climatic Data Center, National Oceanic and Atmospheric Administration, 2013. Washington, DC. <<http://www.ncdc.noaa.gov/>>.
- NCEP, the National Centers for Environmental Prediction, National Oceanic and Atmospheric Administration, 2013. Washington, DC. <<http://www.ncep.noaa.gov/>>.
- NDFD, the National Digital Forecast Database, National Weather Service, NOAA, 2012. Washington, DC. <<http://www.nws.noaa.gov/ndfd/>>.
- NWS, the National Weather Service, NOAA, 2012. Washington, DC. <<http://www.weather.gov>>.
- Olseth, J.A., Skartveit, A., 1984. A probability density function for daily insolation within the temperate storm belts. *Solar Energy* 33, 533.
- Otani, K., Minowa, J., Kurokawa, K., 1997. Study on areal solar irradiance for analyzing areally-totalized PV systems. *Solar Energy Materials and Solar Cells* 47, 281–288.
- Paoli, C., Voyant, C., Muselli, M., Nivet, M.L., 2009. Solar radiation forecasting using ad-hoc time series preprocessing and neural networks. In: *Proc. of 5th International Conference on Intelligent Computing*, Ulsan, South Korea.
- Pfister, G., McKenzie, R.L., Liley, J.B., Thomas, A., Forgan, B.W., Long, C.N., 2003. Cloud coverage based on all-sky imaging and its impact on surface solar irradiance. *Journal of Applied Meteorology* 42, 1421–1434.

- Perez, R., Moore, K., Wilcox, S., Renne, D., Zelenka, A., 2007. Forecasting solar radiation – preliminary evaluation of an approach based upon the national forecast database. *Solar Energy* 81, 809–812.
- Perez, R., Kivalov, S., Schlemmer, J., Hemker, K., Renne, D., Hoff, T.E., 2010. Validation of short and medium term operational solar radiation forecasts in the US. *Solar Energy* 84, 2164–2172.
- Perez, R., Lorenz, E., Pelland, S., Beauharnois, M., Van Knowe, G., Hemker Jr., K., Heinemann, D., Remund, J., Muller, S.C., Traunmuller, W., Steinmauer, G., Pozo, D., Ruiz-Arias, J.A., Fara-Fanego, V., Ramirez-Santigosa, L., Gaston-Romero, M., Pomares, L.M., 2013. Comparison of numerical weather prediction solar irradiance forecasts in the US, Canada, and Europe. *Solar Energy* 94, 305–326.
- Perry, W., 2012. National Weather Service office in Blacksburg, Blacksburg, VA. Private communication.
- Rahim, R., Baharuddin, Mulyadi, R., 2004. Classification of daylight and radiation data into three sky conditions by cloud ratio and sunshine duration. *Energy and Buildings* 36, 660–666.
- Reikard, G., 2009. Predicting solar radiation at high resolutions: A comparison of time series forecasts. *Solar Energy* 83, 342–349.
- Remund, J., Perez, R., Lorenz, E., 2008. Comparison of solar radiation forecasts for the USA. In: *Proc. of 23rd European PV Conference, Valencia, Spain.*
- Robert, C.P., 1995. Simulation of truncated normal variables. *Statistics and computing* 5, 121–125.
- Rousseeuw, P.J., Leroy, A.M., 1987. *Robust regression and outlier detection.* Wiley, New York.

- Şen, Z., 1998. Fuzzy algorithm for estimation of solar irradiation from sunshine duration. *Solar Energy* 63, 39–49.
- Shah, S., Rao, B.M., Kumar, P., Pal, P.K., 2010. Verification of cloud cover forecast with INSAT observation over western India. *Journal of Earth System Science* 119, 775–781.
- Skartveit, A., Olseth, J.A., 1992. The probability density and autocorrelation of short-term global and beam irradiance. *Solar Energy* 49, 447–487.
- Sohne, N., Chaboureau, J.P., Guichard, F., 2008. Verification of cloud cover forecast with satellite observation over West Africa. *Monthly Weather Review* 136, 4421–4434.
- Soubdhan, T., Emilion, R., Calif, R., 2009. Classification of daily solar radiation distributions using a mixture of Dirichlet distributions. *Solar Energy* 83, 1056–1063.
- Sözen, A., Arcaklioğlu, E., Özalp, M., 2004. Estimation of solar potential in Turkey by artificial neural networks using meteorological and geographical data. *Energy Conversion and Management* 45, 3033–3052.
- Suehrcke, H., McCormick, P., 1988. The frequency distribution of instantaneous insolation values. *Solar Energy* 40, 413–422.
- Tovar, J., Olmo, F.J., Alados-Arboledas, L., 1998. One-minute global irradiance probability density distributions conditioned to the optical air mass. *Solar Energy* 62, 387–393.
- Tulcan-Paulescu, E., Paulescu, M., 2008. Fuzzy modelling of solar irradiation using air temperature data. *Theoretical and Applied Climatology* 91, 181–192.
- Udo, S.O., 2000. Sky conditions at Ilorin as characterized by clearness index and relative sunshine. *Solar Energy* 69, 45–53.
- Umemiya, N., Kanou, T., 2008. Classification of sky conditions by the ranges of insolation indices considering CIE standard for general sky. *Light & Visual Environment* 32, 14–19.

- Woyte, A., Belmans, R., Nijs, J., 2007. Fluctuations in instantaneous clearness index: Analysis and statistics. *Solar Energy* 81, 195–206.
- Yan, W., Zhao, S., 2009. GFS-based MOS opaque sky cover and ceiling height for the contiguous United States, Alaska, Hawaii, and Puerto Rico. In: *Proc. of 23rd Conference on Weather Analysis and Forecasting/19th Conference on Numerical Weather Prediction*, Omaha, NE.
- Ye, Q.-Z., Chen, S.-S., 2013. Ultimate meteorological question from observational astronomers: how good is the cloud cover forecast? *Monthly Notices of the Royal Astronomical Society* 428, 3288–3294.
- Zangvil, A., Aviv, O.E., 1987. On the effect of latitude and season on the relation between the diffuse fraction of solar radiation and the ratio of global to extraterrestrial radiation. *Solar Energy* 39, 321–327.
- Zangvil, A., Lamb, P., 1997. Characterization of sky conditions by the use of solar radiation data. *Solar Energy* 61, 17–22.