

H T
W I
G N

Hochschule Konstanz
Fakultät Informatik

Team Project Dijon - Konstanz

Winter term 2022/23

HTWG Konstanz Team



- Pierre Desbruns <Pierre_Desbruns@etu.u-bourgogne.fr>
- Nolane Delille <Nolane_Delille@etu.u-bourgogne.fr>
- Adrien Roy <Adrien_Roy@etu.u-bourgogne.fr>
- Tim Geißendörfer <timgessendoerfer@icloud.com>
- Jens Ullrich <jens.ullrich99@gmail.com>
- Julian Post <ju.post@t-online.de>
- Contact (HTWG, Konstanz): Doris Bohnet (dbohnet@htwg-konstanz.de), O-311
- Contact (ESIREM, Dijon): Gioia Vago (vago@u-bourgogne.fr), Alain Jacquemard (Alain.Jacquemard@u-bourgogne.fr)



Project plan

- **First meeting in Konstanz (6.10.-8.10.22):**
 - getting to know each other
 - Understand more or less the subject
 - define project plan with mile stones, status meetings with supervisors and roles (e.g. who organises online meetings/ assigns tasks?) → keep in mind the time everybody can dedicate to this project!
 - Decide which technologies will be used
 - set up github/ discord channel ...
- **Remote team work:** October 2022 – February 2023
- **Second meeting in Dijon** (*suggestion (HTWG): preferably Sa-Tu: 19-22/11 or Th-Su 1-4/12*): show status
- Final presentation online (before 28/02/2023)

To be
determined !

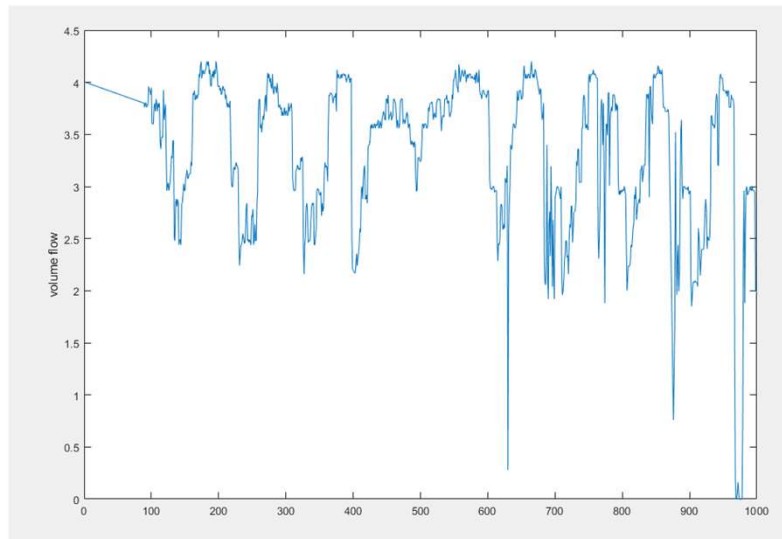
Time series analysis of HVAC sensor data

General Objective: detect automatically if a HVAC system is working normally

1. Bosch Building Technologies provides us with sensor data of a large HVAC-system (heating, ventilation, air, cooling) of an office building (data of last year)
2. **Firstly: Analyse each time series independently**
 1. Decompose into base line and seasonal parts
 2. Use fourier decomposition to predict future values for each time series (hint: you can use neural networks to fit a fourier series to the data or classical fourier analysis tools, both are available e.g. in python)
 3. Identify outliers → what may be a good criteria to detect outliers? Think/ find examples of false positive/ false negative identification when applying your criteria
 4. Assess quality of data → maybe one sensor provides more liable data than another?
3. **Secondly: Analyse causal dependencies between the time series**, e.g. temperature of heating water in/out, temperature of room, outside temperature, performance of heating are intimately related
 1. Find easy causal dependencies (e.g. direct proportionality) between time series (e.g. you could do regression analysis on pairs of time series to detect linear dependencies)
 2. Can you use these dependencies to identify outliers in the time series? Maybe combine this criterion with 2.3.?
4. **Thirdly: Build a user-friendly (web)application**
 1. User input: time series, name/SI unit, maybe rules/reasons
 2. Output: Show time series and results of analysis (e.g. prediction, outliers)

Data

- Data = sensor data of HVAC (heating, ventilation, air conditioning) of office building
- Temperature (setpoint/ actual value), volume flow, flow temperature, valves, power etc. with timestamp of the last year (more or less); one data point each 15 minutes



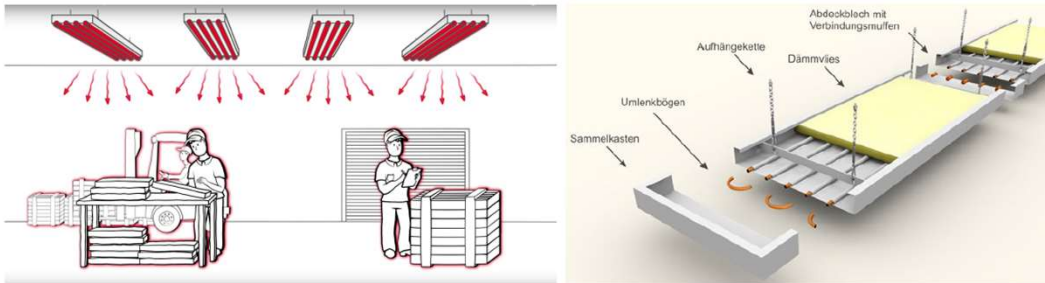
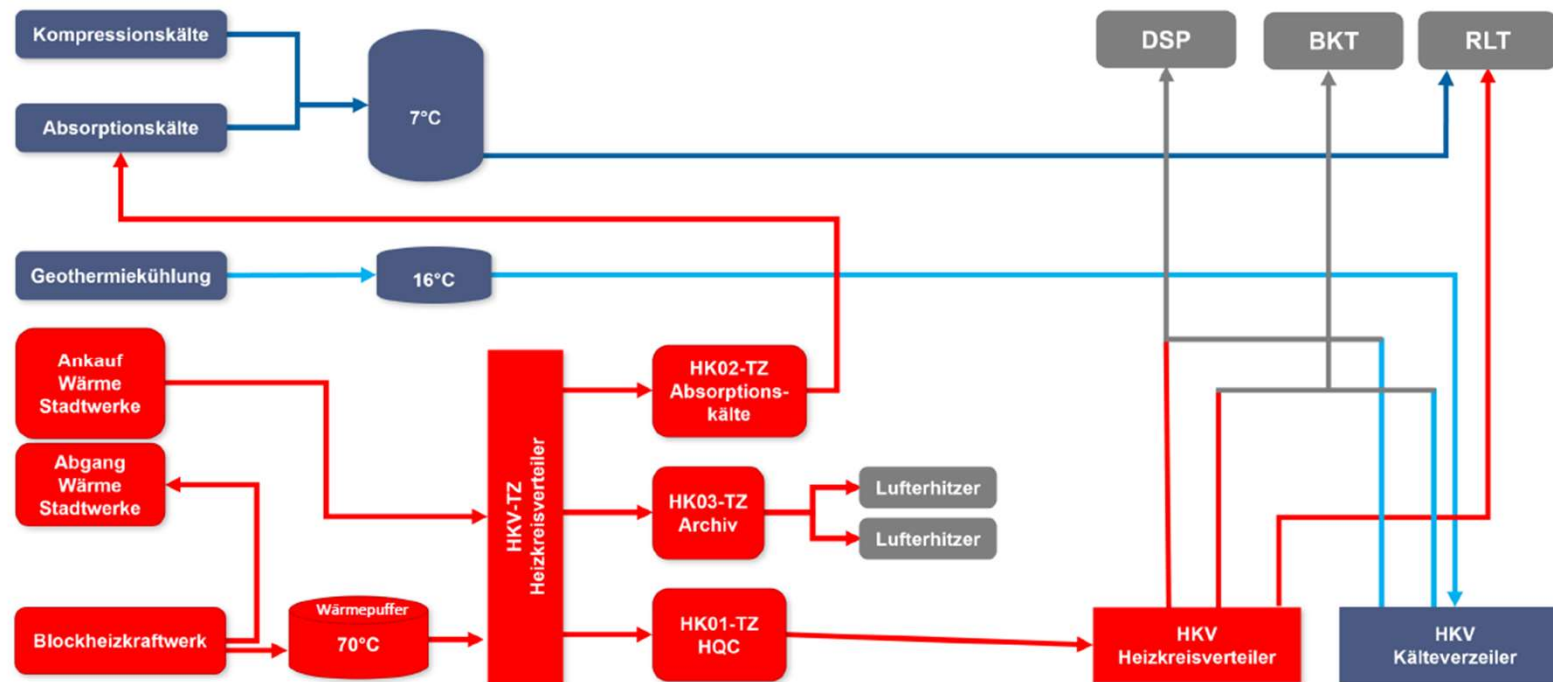


Abbildung 3: Deckenstrahlplatte Funktionsweise [15]

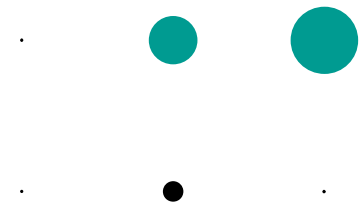
Abbildung 4: Schnittdarstellung Deckenstrahlplatten [16]

- DSP Deckenstrahlplatte
- BKT Betonkernaktivierung
- RLT Raum-Luft-Technik



Standard way to analyse

- Decomposition of signal into base line ($=a_0 t$) and periodic components ($=a_0 \sin(\omega_0 t) + a_1 \sin(\omega_1 t) + \dots$)
- Typical periods in our case: one day (=96 data points), one week (=672), one year (35040 data points)
- How to find decomposition?
 - Given data points (t_i, y_i) find coefficients a_0, a_1, \dots , such that the distance $\sum |a_0 t_i + a_0 \sin(\omega_0 t_i) + a_1 \sin(\omega_1 t_i) + \dots - y_i|$ is minimal. There are different ways to identify frequencies and amplitudes (e.g. by optimization /neural network or fourier analysis)



Bewertungskriterien (HTWG)

Projektmanagement/ Organisation des Teams (40%)

Kriterien:

1. Es gibt einen Projektplan, der klar definierte Arbeitspakete und Meilensteine enthält
2. Aufgaben werden mit Hilfe eines Repository innerhalb des Teams verteilt
3. Termine werden eingehalten, bei Verzögerungen wird rechtzeitig reagiert und notfalls der Plan angepasst
4. Kommunikation/ Abstimmung (jeder kennt immer den Stand des Projekts und die eigene Aufgabe)

Umsetzung (40%)

Kriterien:

1. Funktionalität (Ziele des Projektplans erreicht)
2. Wahl von Programmiersprache/ Tools waren den Anforderungen angemessen
3. sauberer Code (dokumentiert, einheitlicher Stil, sinnvolle Strukturierung...)

Abschlusspräsentation (20%)

Kriterien:

1. Gemeinsame Präsentation ca. 30-45 min zum Projektabschluss (Termin kann von Team festgelegt werden: Ende Januar – Ende Februar)
2. Vorgehen & Ergebnisse vorstellen
3. Was hat gut geklappt? Was könnte man das nächste Mal besser machen?

Note: Es wird eine Team-Note vergeben. Dem Team steht es frei, innerhalb des Teams die Note zu differenzieren und einem einzelnen Teammitglied z.B. eine bessere Note, einem andere eine etwas schlechtere zu vergeben. Im Durchschnitt müssen die Einzelnoten die Team-Note ergeben.