

ČESKÉ VYSOKÉ  
UČENÍ TECHNICKÉ  
V PRAZE

FAKULTA  
STROJNÍ



DIPLOMOVÁ  
PRÁCE

2021

JAN  
DYNTAR

## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Dyntar** Jméno: **Jan** Osobní číslo: **467312**  
Fakulta/ústav: **Fakulta strojní**  
Zadávající katedra/ústav: **Ústav mechaniky tekutin a termodynamiky**  
Studijní program: **Aplikované vědy ve strojním inženýrství**  
Specializace: **Aplikovaná mechanika**

## II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

**Využití dopředné neuronové sítě k predikci výkonu větrné elektrárny a k odhadu spotřeby tepla objektu**

Název diplomové práce anglicky:

**Utilization of back propagation neural network to predict wind power and to guess heat consumption of building**

Pokyny pro vypracování:

S využitím poskytnuté literatury stručně popište rekonstrukci fázového prostoru chaotického systému. Při popisu uveďte funkce dostupné v programu MATLAB. Zaměřte se na analýzu chaotické časové řady získané měřením rychlosti větru. Časovou řadu analyzujte z pohledu potenciálu pro výrobu elektrické energie za použití větrné elektrárny. Proveďte predikci budoucího vývoje rychlosti větru. K predikci použijte parametry rekonstruovaného fázového prostoru a dopřednou neuronovou síť implementovanou v prostředí MATLAB. V dalším kroku analyzujte podobným způsobem časovou řadu týkající se spotřeby tepla vybraného objektu a pokuste se predikovat budoucí spotřebu tepla. Získané výsledky podrobně diskutujte. Rozsah průvodní zprávy: cca 50 stran

Seznam doporučené literatury:

dle doporučení vedoucího práce

Jméno a pracoviště vedoucí(ho) diplomové práce:

**doc. Ing. Tomáš Hyhlík, Ph.D., ústav mechaniky tekutin a termodynamiky FS**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **26.10.2021**

Termín odevzdání diplomové práce: **07.01.2022**

Platnost zadání diplomové práce: \_\_\_\_\_

\_\_\_\_\_  
doc. Ing. Tomáš Hyhlík, Ph.D.  
podpis vedoucí(ho) práce

\_\_\_\_\_  
Ing. Michal Schmirler, Ph.D.  
podpis vedoucí(ho) ústavu/katedry

\_\_\_\_\_  
prof. Ing. Michael Valášek, DrSc.  
podpis děkana(ky)

## III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

\_\_\_\_\_  
Datum převzetí zadání

\_\_\_\_\_  
Podpis studenta

## **Anotace**

Jméno autora:	Jan Dyntar
Název diplomové práce:	Využití dopředné neuronové sítě k predikci výkonu větrné elektrárny a k odhadu spotřeby tepla objektu
Anglický název:	Utilization of feedforward neural network to predict wind power and to guess heat consumption of building
Akademický rok:	2021/2022
Ústav:	U 12112 – Ústav mechaniky tekutin a termodynamiky
Vedoucí práce:	doc. Ing. Tomáš Hyhlík, Ph.D.
Bibliografické údaje:	Počet stran: 66 Počet obrázků: 29 Počet tabulek: 7 Počet příloh: 0

Klíčová slova: rekonstruovaný fázový prostor, dopředná neuronová síť, predikce výkonu větrné elektrárny, predikce spotřeby tepla

Keywords: reconstructed phase space, feedforward neural network, wind power prediction, heat consumption prediction

### Abstrakt:

Tato diplomová práce se zabývá využitím dopředných neuronových sítí k predikci chaotických časových řad. K tomuto účelu byla použita data rychlosti větru a tepla odebíraného ze systému dálkového vytápění. Cílem práce bylo analyzovat časovou řadu rychlosti větru z hlediska potenciálu pro výrobu elektrické energie, provést predikci budoucího vývoje rychlosti větru a pokusit se predikovat spotřebu tepla objektu. Z časových řad byly v prostředí programu MATLAB rekonstruovány fázové prostory, jež byly následně pomocí dopředné neuronové sítě implementované v témž programu, použity pro predikci. Predikce obou časových řad byla úspěšně. Podařilo se predikovat několik budoucích hodnot, anebo alespoň sledovat trend jejich vývoje.

### Abstract:

This master's thesis deals with utilization feedforward neural network for chaotic time series prediction. Wind speed data and data of heat consumed from heating network were used for this purpose. Goals of this thesis were to analyse chaotic wind speed time series in terms of potential for electricity generation using wind turbine, to predict future development of wind speed and to attempt to predict heat consumption of building. I reconstructed phase spaces from these time series in environment of the program MATLAB. Then they were used for prediction using feedforward neural network implemented in the program. Prediction of both time series was successfully carried out. It

was accomplished to correctly predict several values or at least follow development of chart trend.

„Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré zdroje, ze kterých jsem čerpal, jsou uvedeny v seznamu použité literatury. Práce nebyla použita k získání jiného ani stejného titulu.“

Datum:

.....

Jan Dyntar

**Poděkování:**

Rád bych poděkoval doc. Ing. Tomáši Hyhlíkovi, Ph.D. za vedení práce, cenné rady, věnovaný čas a trpělivost při vypracování diplomové práce.

Děkuji také svojí manželce, rodině a přátelům za velkou podporu při tvorbě této práce.

## Obsah

1. Seznam zkratk a symbolů.....	7
2. Úvod.....	9
3. Cíle práce .....	11
4. Předpoklady a metody řešení .....	12
4.1. Výkon větrné elektrárny.....	12
4.1.1. Rychlost větru vstupujícího do rotoru větrné elektrárny .....	13
4.1.2. Výkonová křivka turbíny .....	15
4.2. Předpověď rychlosti větru.....	16
4.3. Fázový prostor.....	17
4.3.1. Rekonstrukce fázového prostoru.....	17
4.3.2. Určení časového zpoždění .....	18
4.3.3. Určení dimenze vnoření.....	20
4.3.4. Rekonstrukce fázového prostoru v programu MATLAB .....	23
4.4. Neuronové sítě.....	24
4.4.1. Učení ANN.....	25
4.4.2. Druhy ANN .....	26
4.4.3. Pozice ANN vůči jiným výpočetním modelům .....	27
4.4.4. Neuronové sítě v MATLAB .....	27
5. Vlastní práce – rychlost větru .....	29
5.1. Zpracování a analýza dat rychlosti větru .....	29
5.2. Výkon větrné elektrárny.....	31
.....	34
5.3. Analýza časové řady pro učení neuronové sítě.....	35
5.4. Proces učení neuronové sítě .....	37
5.5. Predikce.....	40
6. Vlastní práce – spotřeba tepla .....	42
6.1. Data pro výpočet spotřeby tepla .....	42
6.2. Predikce tepelného výkonu.....	46

7.	Výsledky a diskuse .....	48
7.1.	Predikce rychlosti větru .....	48
7.2.	Predikce tepelného výkonu.....	54
8.	Závěr .....	58
9.	Literatura .....	60
10.	Seznam obrázků.....	62
11.	Seznam tabulek.....	64



## 1. Seznam zkratek a symbolů

### zkratky:

ANN	Umělá neuronová síť (Artificial neural network)
BR	Bayesianova regularizace
FFNN	Dopředná neuronová síť (Feedforward neural network)
FBNN	Zpětná neuronová síť (Feed-backward neural network)
LM	Levenberg-Marquardt
NN	Neuronová síť (Neural network)

### symboly:

$a$	[1]	poměr vzdáleností stavů fázového prostoru
$A$	[1]	autokorelační funkce
$b$	[1]	práh neuronu
$c_k$		odhad autokovarianční funkce (jednotky podle proměnné funkce)
$c_p$	[J/kg·K]	měrná tepelná kapacita
$E$	[1]	průměr poměrů vzdáleností stavů fázového prostoru
$E_k$	[J]	kinetická energie
$I$	[1]	míra vzájemné informace
$m$	[kg]	hmotnost
$M$	[1]	počet vektorů rekonstruovaného fázového prostoru
$\dot{m}$	[kg/s]	hmotnostní tok
$m_d$	[1]	dimenze vnoření
$P$	[W]	výkon
$P(x)$	[1]	pravděpodobnost výskytu hodnoty $x$
$P_q$	[W]	teplotní výkon

Q	[J]	teplo
R	[1]	vzdálenost stavů fázového prostoru
s	[1]	počet intervalů v časové řadě
S	[m <sup>2</sup> ]	průřez
t	[s]	čas
T	[°C]	teplota
$\tau$	[1]	časové zpoždění
v	[m/s]	rychlost větru
V	[m <sup>3</sup> ]	objem
$\dot{V}$	[m <sup>3</sup> /s]	průtok
$v_{ref}$	[m/s]	rychlost větru ve výšce měření
$v_t$	[m/s]	rychlost větru ve výšce turbíny
w	[1]	váha spoje neuronů
x		proměnná (jednotky závisí na druhu proměnné)
$X_m$	[1]	stav fázového prostoru
z	[m]	výška turbíny
$z_0$	[m]	délka drsnosti
$z_{ref}$	[m]	výška měření
$\theta$	[1]	Heavisideova funkce
$\rho$	[kg/m <sup>3</sup> ]	hustota
$\sigma$	[1]	směrodatná odchylka v procentech
$\sigma^2(v)$	[m <sup>2</sup> /s <sup>2</sup> ]	střední kvadratická odchylka rychlosti

## 2. Úvod

Schopnost předpovídat budoucnost byla vždy velmi ceněna, a protože se lidstvo postupem času přesunulo od věštění a vykládání karet přesunulo k výpočtům podloženým teorií i praxí, dosahuje stále přesnějších předpovědí toho, co se teprve stane. V různých směrech však dosahujeme různých výsledků. Zatímco v některých oblastech, jako například pohyb hvězd a planet, jsou naše schopnosti předpovídat budoucí vývoj téměř bezchybné, v jiných případech, jako je například pohyb částice v turbulentním proudění, dokážeme o budoucím vývoji pouze spekulovat. Jedním z nejoblíbenějších témat zabývajícím se predikcí budoucnosti je nepochybně předpověď počasí. I ta už si za historii lidstva prošla velmi významným pokrokem, ale i přes nejmodernější numerické metody nedosahuje takové přesnosti, jak by si lidstvo přálo. Na počasí totiž působí takové množství vnějších vlivů, že je ani ty nejmodernější počítače nedokážou spolehlivě popsat a musí se spoléhat na zjednodušené výpočty, zkušenosti meteorologů a statistické výpočty.

Ve druhé polovině minulého tisíciletí byl objeven takový výpočetní model, který měl potenciál překonat i takto náročné problémy. Tento výpočetní systém se nazývá umělá neuronová síť (dále pouze ANN podle anglického názvu Artificial neural network) a jak už vyplývá z názvu, byl silně inspirován funkcí biologické neuronové sítě, která tvoří základ nervové soustavy člověka. Od doby svého objevení prošly ANN velkým pokrokem a rozvojem a našly uplatnění v mnoha oblastech, jednou z nejvýznamnějších je například rozpoznávání obrazu. Stále je však mnoho prostoru pro další výzkum, stále existují cesty, jak zlepšit schopnosti a možnosti použití ANN a limity se zdají být v nedohlednu.

Tato práce se pokouší tento potenciál využít a s pomocí jednoduchých neuronových sítí dokázat předpovídat jednu z nejvýznamnějších složek počasí, a to rychlost větru. Tato znalost v poslední době totiž nabývá na stále větší důležitosti. Nutnost redukovat emise a vypouštění uhlíku do atmosféry tlačí energetický průmysl k vyhledávání alternativ za populární a poměrně levné spalování uhlí. Jednou z nejoblíbenějších cest pro tvorbu tzv. čisté energie je právě využívání větrných elektráren. Problémem tohoto způsobu vytváření elektřiny je jeho do určité míry nespolehlivost. Množství získané energie totiž vždy závisí na aktuální rychlosti větru. Znalost, jaký je budoucí vývoj rychlosti větru, by tak energetickému průmyslu mohla umožnit se připravit na přebytek či nedostatek vyprodukované elektřiny a připravit systémy, které by ho kompenzovaly. První částí této práce se zabývá snahou tento budoucí vývoj predikovat. Z dat rychlosti větru, která se nacházejí ve formě chaotické časové řady, je v prostředí programu MATLAB rekonstruován fázový prostor a pomocí

dopředné ANN prováděna samotná predikce. Přitom je zde také řešen způsob, jak z naměřených rychlostí větru určit výkon turbíny větrné elektrárny.

Druhá část práce se věnuje další oblasti, kde by byla schopnost predikovat budoucí vývoj též velmi ceněna. Touto problematikou je spotřeba tepla objektu. Z naměřených teplot a průtoků je určován tepelný výkon, z něhož lze vypočítat množství tepla odebíraného z topného systému. Na fázový prostor sestavený z časové řady tepelného výkonu je obdobným způsobem aplikována neuronová síť, která provádí predikci budoucího vývoje spotřeby tepla budovy. Schopnost znát předem spotřebu tepla různých objektů by mohlo vést k významným energetickým úsporám, bylo by totiž možné předem znát přesné množství tepla, které je třeba dodat. Tím by se snížilo množství nadbytečného tepla a tepla unikajícího do okolního prostředí.

### 3. Cíle práce

- stručně popsat rekonstrukci fázového prostoru chaotického systému a uvést funkce dostupné v programu MATLAB
- analyzovat chaotickou časovou řadu získanou měřením rychlosti větru z pohledu potenciálu pro výrobu elektrické energie za použití větrné elektrárny
- provést predikci budoucího vývoje rychlosti větru, k tomu použít parametry rekonstruovaného fázového prostoru a dopřednou neuronovou síť implementovanou v prostředí programu MATLAB
- podobným způsobem analyzovat časovou řadu z hlediska spotřeby tepla vybraného objektu a pokusit se predikovat budoucí spotřebu tepla

#### 4. Předpoklady a metody řešení

Hlavním cílem této práce bylo provést predikci budoucího vývoje rychlosti větru. Dalším úkolem bylo analyzovat data ve formě chaotické časové řady sestavit v prostředí programu MATLAB rekonstruovaný fázový prostor a na něj aplikovat dopřednou neuronovou síť. Cílem bylo dosáhnout predikce, která bude alespoň několik hodin sledovat trend budoucího vývoje rychlosti větru. V dalším kroku měla být obdobným způsobem provedena predikce spotřeby tepla vybraného objektu. Zároveň bylo i úkolem analyzovat naměřená data rychlosti větru z hlediska potenciálu pro výrobu elektrické energie větrnou elektrárnou.

##### 4.1. Výkon větrné elektrárny

Principem větrné elektrárny je přeměna kinetické energie větru na mechanickou energii rotoru, která je následně turbínou přeměněna na elektrickou energii. Pro výpočet výkonu turbíny lze tento problém zjednodušit na pouze dvě proměnné. První proměnnou je kinetická energie větru a druhou je účinnost, s jakou ji dokáže elektrárna přeměnit na výkon. Kinetická energie větru  $E_k$  závisí na jeho rychlosti  $v$  a hmotnosti  $m$ , jak vyplývá ze vzorce:

$$E_k = \frac{1}{2} \cdot m \cdot v^2 \quad (4.1.1)$$

Když jsou obě strany rovnice poděleny časem, vyjde výraz pro výkon větru:

$$P = \frac{1}{2} \cdot \dot{m} \cdot v^2 \quad (4.1.2)$$

kde  $\dot{m}$  je hmotnostní tok větru. Ten lze určit z výrazu:

$$\dot{m} = \rho \cdot \dot{V} \quad (4.1.3)$$

$$\dot{m} = \rho \cdot S \cdot v \quad (4.1.4)$$

kde  $\rho$  je hustota vzduchu,  $\dot{V}$  je objemový průtok vzduchu a  $S$  je průřez proudu vzduchu. Dosazením do vzorce pro výkon pak vychází výraz:

$$P = \frac{1}{2} \cdot \rho \cdot S \cdot v^3 \quad (4.1.5)$$

Průřez je konstantní, jedná se totiž o plochu rotoru elektrárny. Hustota vzduchu se výrazně nemění a její změna nemá na výkon tak velký vliv jako rychlost větru. Proto je v této práci

uvažována jako konstantí. Jak už bylo řečeno, rychlost větru má na výkon výrazný vliv, proto se jí budu věnovat v následující kapitole.

#### **4.1.1. Rychlost větru vstupujícího do rotoru větrné elektrárny**

Rychlost větru se výrazně mění v závislosti na vzdálenosti od zemského povrchu, proto by vždy bylo ideální, měřit ji ve stejné výšce (výškou bude dále myšlena vzdálenost od zemského povrchu), jako se má nacházet střed rotoru větrné elektrárny. Často tomu tak však není a v tom případě musí být naměřená rychlost přepočítána na rychlost v požadované výšce. Hlavní příčinou snižování rychlosti větru s klesající výškou je podle [8] ovlivňování vzdušných proudů zemským povrchem. Tento efekt je možné rozdělit na tři oblasti, a to na vliv překážek, vliv tvaru krajiny a drsnost povrchu.

Překážky zpomalují proud vzduchu, protože je musí obtéci. Míra zpomalení závisí na tvaru a velikosti překážky. Ovlivněna však není pouze ta část proud, v jejímž směru se překážka nachází. Obtékající proud se mísí s okolním vzduchem a ten je z toho důvodu také zpomalován. Pro větrné elektrárny je proto žádoucí, aby se nacházeli v dostatečné výšce či vzdálenosti od objektů, které by mohly zpomalovat vzduch vstupující do rotoru. Podle [8] se za takovou dostatečnou vzdálenost považuje 1 kilometr. Vzdálenější překážky už na rychlost větru v měřeném místě mají zanedbatelný vliv.

Další oblastí je vliv tvaru krajiny. Příkladem tohoto vlivu je tzv. tunelový efekt. Dochází k němu například mezi dvěma vysokými budovami, v horském průsmyku nebo i mezi dvěma blízkými kopci. V takových případech je totiž vzduch hnán do mezery, která je jediným místem, kudy může proudit dál. Směrování velkého množství vzduchu do jednoho místa vytváří vyšší tlak, který způsobuje značný nárůst rychlosti větru. Na podobném principu také funguje tzv. kopcový efekt, kdy je vzduch stlačován a tím urychlován při přechodu přes hřeben kopce. Tyto efekty jsou naopak vyhledávanými fenomény, protože větrné turbíně umožňují dosahovat daleko vyšších výkonů, než by bylo za běžných podmínek možné.

Třetí oblastí je drsnost povrchu. V tomto případě se jedná o efekt zpomalování proudění vzduchu třením o zemský povrch. Velký vliv na míru zpomalení má prostředí, v jakém se nachází zkoumané místo. Ty se rozlišují do tříd značených hodnotami 0 až 4, viz. Tabulka 1. Do třídy 0 patří nejméně ovlivňující prostředí jako například vodní povrch a do

třídy 4 ty s největším vlivem jako například velká města s mnoha vysokými budovami. Každé třídě je přiřazována tzv. délka drsnosti. Ta udává, v jaké výšce je teoreticky vítr vlivem prostředí zpomalen až na nulovou rychlost.

Pokud je třeba určit rychlost větru v jiné výšce, než v jaké byla naměřena, musí být do výpočtu zahrnuta drsnost povrchu. Ten má tvar:

$$v_t = v_{ref} \cdot \frac{\ln\left(\frac{z}{z_0}\right)}{\ln\left(\frac{z_{ref}}{z_0}\right)} \quad (4.1.1.1)$$

kde  $v_t$  je rychlost větru ve výšce turbíny,  $v_{ref}$  je měřená rychlost,  $z$  je výška turbíny,  $z_0$  je délka drsnosti a  $z_{ref}$  je výška, ve které byla rychlost měřena. Tento výraz platí za předpokladu, že zemský povrch v daném místě není na rozdíl od vzduchu nějakým způsobem ochlazován či zahříván.

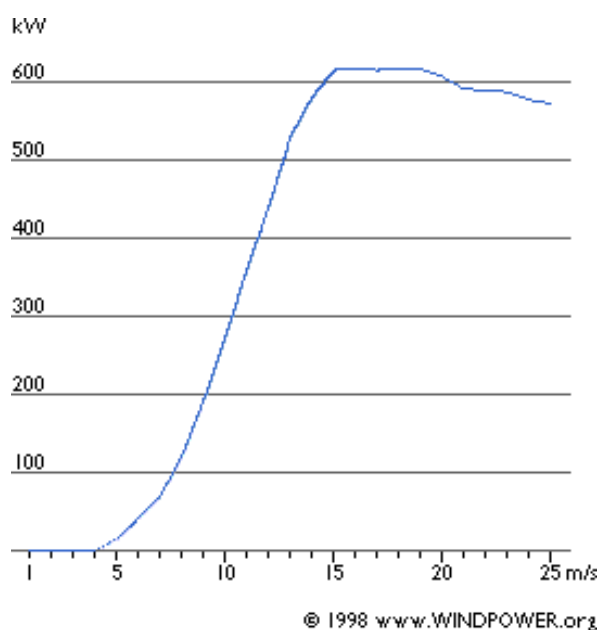
Třída drsnosti	Délka drsnosti [m]	Typ krajiny
0	0,0002	Vodní povrch
0,5	0,0024	Úplně otevřený terén s hladkým povrchem, např beton, posekaný trávník, atd.
1	0,03	Otevřené zemědělské oblasti bez plotů a mezí, pouze mírné kopce
1,5	0,055	Zemědělská půda s několika budovami a porostlými mezemi ve vzdálenosti přibližně 1250 metrů
2	0,1	Zemědělská půda s několika budovami a porostlými mezemi ve vzdálenosti přibližně 500 metrů
2,5	0,2	Zemědělská půda s mnoha budovami a porostlými mezemi ve vzdálenosti přibližně 250 metrů
3	0,4	Vesnice, malá města, lesy a velmi drsný a nerovný terén
3,5	0,8	Města s vysokými budovami
4	1,6	Velkoměsta s vysokými budovami a mrakodrapy

Tabulka 1: Tabulka zobrazující třídy drsnosti a hodnoty délek drsnosti k příslušným typům krajiny převzato a upraveno z [8]



#### 4.1.2. Výkonová křivka turbíny

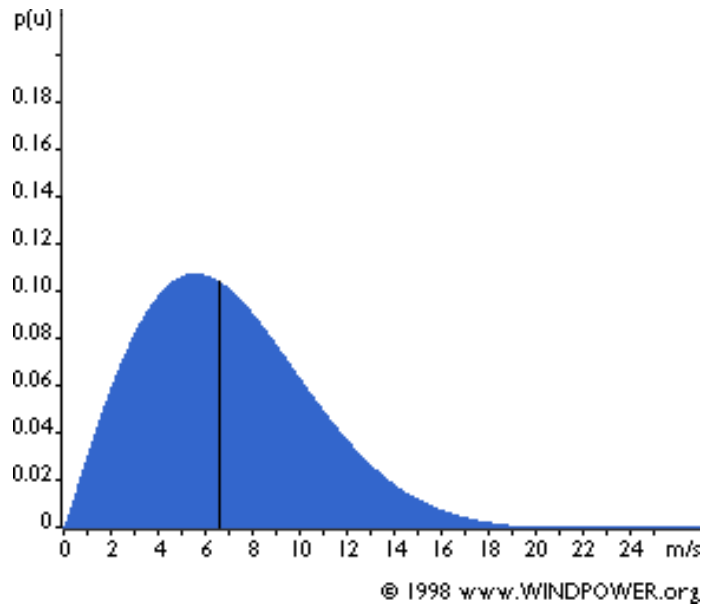
Vztah mezi rychlostí větru a výkonem turbíny se vyjadřuje pomocí výkonové křivky. Tento vztah zahrnuje účinnost elektrárny i průměr rotoru. Na vodorovné ose tohoto grafu je zobrazena rychlost větru a na svislé ose odpovídající výkon turbíny viz Obrázek 1. Hodnoty těchto křivek většinou poskytují sami výrobci turbín. V ideálním případě by měli být sestaveny různé výkonové křivky pro různé hustoty vzduchu, ale často tomu tak není. Tyto grafy se sestavují měřením v terénu, kdy jsou zaznamenávány hodnoty rychlosti větru a k nim jsou přiřazovány naměřené hodnoty výkonu turbíny v odpovídajících časech. Rychlost větru je podle [9] třeba měřit co nejbližně turbíně, aby hodnoty co nejvíce odpovídaly realitě, nesmí to však být až moc blízko, protože rotor turbíny ve svém okolí vytváří turbulence, které by naměřené rychlosti ovlivňovaly. Ideální je anemometr umístit ve vzdálenosti odpovídající přibližně 2,5násobku průměru rotoru.



Obrázek 1: Výkonová křivka turbíny, na vodorovné ose je zobrazována rychlost větru a na svislé ose výkon turbíny [8]

Na reálnou výkonovou křivku turbíny má podle [9] kromě hustoty vzduchu vliv i mnoho dalších faktorů. Mezi ty nejvýznamnější patří stříh větru, turbulence a sklon proudění vzduchu. Tyto efekty se však často mění a lze je tak vzít v potaz pouze jako průměrné hodnoty pomocí rozsáhlejších měření. Další chybu mezi teorií a realitou může způsobit fakt, že výpočty uvažují pouze s rychlostí větru ve výšce osy rotoru. Zvláště u rotorů

o velkých průměrech se ale mohou větrné podmínky působící na horní a spodní část rotoru měnit natolik, že způsobují patrné rozdíly mezi očekávaným a skutečným výkonem turbíny.



Obrázek 2: Weibullův graf, na svislé ose je zobrazena rychlost větru a na svislé ose pravděpodobnost, že vítr bude danou rychlostí foukat [8]

Výpočet výkonu větrné elektrárny se podle [8] provádí následujícím způsobem. Nejprve je nutné sestavit Weibullův graf. Ten zobrazuje pravděpodobnost, jakou rychlostí vítr v měřeném místě vane. Na vodorovné ose tedy leží rychlost větru a na svislé ose je pravděpodobnost, že bude vítr foukat danou rychlostí viz Obrázek 2. Následně jsou hodnoty pravděpodobnosti z Weibullova grafu pro každou rychlost roznášeny odpovídající hodnotou z výkonové křivky. Součtem výsledných hodnot je průměrný výkon dané větrné turbíny.

#### 4.2. Předpověď rychlosti větru

Předikce rychlosti větru je v dnešní době nejčastěji prováděna jako součást předpovědi počasí. Ta je podle [1] prováděna třemi základními způsoby. První metodou je tzv. synoptická předpověď. Při použití této metody jsou nejprve meteorologickými stanicemi shromážděna data, ze kterých jsou sestaveny synoptické mapy. Tyto mapy jsou následně analyzovány odborníkem, který na základě vlastních znalostí a zkušeností sestaví

předpověď budoucího vývoje počasí. Druhá metoda se nazývá numerická předpověď. Oblast, pro kterou je počasí předpovídáno, je rozdělena na síť bodů. Pro každý bod je následně pomocí systémů diferenciálních rovnic výkonnými počítači proveden výpočet. Z jejich výsledků je poté sestavena výsledná předpověď. Třetí metoda se nazývá statistická. Ta sestavuje předpověď na základě matematické statistiky a teorie pravděpodobnosti. Využívá data zaznamenané během historie a snaží se předpovědět budoucí vývoj na základě vývoju podobných situací v minulosti. V současnosti jsou většinou při sestavování předpovědi kombinovány všechny tři metody, aby bylo dosaženo co nejvyšší přesnosti.

Už na konci minulého tisíciletí se však začaly objevovat snahy použít pro predikování rychlosti větru neuronové sítě. Například v práci [10] je napsáno, že se její autoři pokusili provést predikci pro průměrné měsíční a denní hodnoty rychlosti větru. Dle výsledků práce byli úspěšní. Neuronové sítě totiž podávali přesnější výsledky než autoregresivní model, který pro srovnání aplikovali na stejnou sadu dat. Od té doby zaznamenaly výpočetní schopnosti počítačů i neuronové sítě výrazný posun. Mělo by tak být možné provádět přesnější a dlouhodobější predikce.

### **4.3. Fázový prostor**

Naměřené hodnoty rychlosti větru tvoří časovou řadu čili vektor o rozměrech  $1 \times n$ . Ve snaze provést predikci pomocí tohoto vektoru, by klasická dopředná neuronová síť nemohla být úspěšná. Pro předpověď by jí totiž byla jako vstup dodána pouze aktuální hodnota bez jakékoliv znalosti předchozího vývoje. Tento problém je možné vyřešit rekonstrukcí fázového prostoru z dat časové řady. V tomto případě bude poskytnuto neuronové síti více vstupních dat na jejichž základě bude predikci provádět.

#### **4.3.1. Rekonstrukce fázového prostoru**

Rekonstrukce fázového prostoru se nejčastěji provádí Metodou časových zpoždění. Ta probíhá tak, že nejprve máme časovou řadu  $x(t_i)$ , kde  $i = 1, 2, \dots, N$ ,  $t_i$  jsou rovnoměrné časové kroky a  $x$  je měřená veličina. Fázový prostor se pak podle [4] sestaví ve formě vektorů:

$$X(t_1) = [x(t_1), x(t_1 + \tau), \dots, x(t_1 + (m_d - 1)\tau)]$$

$$X(t_2) = [x(t_2), x(t_2 + \tau), \dots, x(t_2 + (m_d - 1)\tau)]$$

...

$$X(t_k) = [x(t_k), x(t_k + \tau), \dots, x(t_k + (m_d - 1)\tau)]$$

, kde  $k = 1, \dots, M$ ,  $\tau$  je časové zpoždění,  $m_d$  je dimenze vnoření a  $M = N - (m_d - 1) \cdot \tau$  je počet vektorů (stavů) rekonstruovaného fázového prostoru. Nejdříve je však nutné zvolit takové  $m$  a  $\tau$ , aby bylo možné dosáhnout požadovaného výsledku. Pokud by bylo časové zpoždění zvoleno příliš malé, projevil by se v hodnotách náhodný šum, a kdyby bylo naopak zvoleno příliš velké, rekonstruovaný signál nebude reprezentovat dynamiku časové řady.

#### 4.3.2. Určení časového zpoždění

Existuje několik metod pro určení časového zpoždění  $\tau$ . Jednou z nich je Metoda určení autokorelační funkce  $A(k)$ , která se podle [4] provádí následujícím způsobem:

$$A(k) = \frac{c_k}{c_0} \quad (4.3.2.1)$$

kde  $c_k$  je odhad autokovarianční funkce a vyjádří se jako:

$$c_k = \frac{1}{N} \sum_{i=1}^{N-k} [x(t_i) - \bar{x}][x(t_{i+k}) - \bar{x}] \quad (4.3.2.2)$$

kde  $\bar{x}$  je aritmetický průměr hodnot časové řady.

Časové zpoždění  $\tau$  pak odpovídá takové hodnotě  $k$ , kdy  $A(k)$  klesne pod určitou zvolenou hodnotu. Místo autokorelační funkce 1. řádu je pro hledání ideálního  $\tau$  možné použít také funkce vyšších řádů, nebo i nelineární autokorelační funkci.

Dalším způsobem určení  $\tau$  je podle [4] Metoda míry vzájemné informace I. Pro míru vzájemné informace mezi řadou  $x(t_i)$  a posunutou řadou  $x(t_i + \tau)$  platí:

$$I(\tau) = \sum_{x(t_i)x(t_i+\tau)} P(x(t_i), x(t_i + \tau)) \cdot \log_2 \left[ \frac{P(x(t_i), x(t_i + \tau))}{P(x(t_i)) \cdot P(x(t_i + \tau))} \right] \quad (4.3.2.3)$$

kde  $P(x(t_i))$  je pravděpodobnost výskytu hodnot  $x(t_i)$  v daném intervalu časové řady,  $P(x(t_i + \tau))$  je pravděpodobnost výskytu hodnot  $x(t_i + \tau)$  v dalším dané intervalu při posunutí v čase o časové zpoždění  $\tau$  a  $P(x(t_i), x(t_i + \tau))$  je sdružená pravděpodobnost výskytu hodnot  $x(t_i)$  v prvním intervalu a zároveň jejich výskyt při posunutí o  $\tau$  ve druhém intervalu. Množství intervalů v časové řadě, ze kterých se má počítat pravděpodobnost, lze určit např. pomocí Sturgesova pravidla, kde je počet intervalů značen jako  $s$ :

$$s = 1 + [\log_2 \cdot N] \quad (4.3.2.4)$$

Jako ideální  $\tau$  se pak většinou bere hodnota, při které míra vzájemné informace dosáhne prvního minima.

Třetí metodou pro určení časového zpoždění  $\tau$ , kterou zde zmíním, je tzv. C-C metoda. Při jejím popisu vycházím z informací z [5]. Její hlavní myšlenkou je rozdělit časovou řadu  $\{x_1, x_2, \dots, x_M\}$  do  $p$  nezávislých časových řad, kde  $p$  je index zpoždění. Dalším krokem je výpočet  $S_d$  z vytvořených časových řad, což se provádí následujícím způsobem.  $\theta$  je Heavisidova funkce neboli jednotkový skok. Dosadí se do vzorce:

$$C = \frac{2}{N} (N - 1) \sum_{1 \leq i < j \leq N} \theta(r - |x_i - x_j|) \quad (4.3.2.5)$$

kde platí, že  $N = M - (m_d - 1) \cdot \tau$  a  $r > 0$ , přičemž hodnoty  $m_d$  a  $r$  se pohybují v rozmezích:

$$2 \leq m_d \leq 5 \quad (4.3.2.6)$$

$$\frac{\sigma}{2} \leq r \leq 2 \cdot \sigma \quad (4.3.2.7)$$

kde  $\sigma$  je standartní odchylka řešených sad dat. Proměnná  $C$  se následně dosazuje do vzorce:

$$S_d = \frac{1}{p} \sum_{s=1}^p [C - C^{m_d}] \quad (4.3.2.8)$$

Poté se určí maximální rozmezí hodnot proměnné S:

$$\Delta S_d = \max(S_d) - \min(S_d) \quad (4.3.2.8)$$

které se dosadí do vzorce:

$$\Delta \bar{S}_d = \frac{1}{4} \sum_{m=2}^5 \Delta S \quad (4.3.2.9)$$

Čas, ve kterém je poté určeno dosažení prvního lokálního minima  $\Delta \bar{S}_d$  je označen za hodnotu optimálního časového zpoždění  $\tau$ .

#### 4.3.3. Určení dimenze vnoření

Další hodnotou, kterou je třeba určit pro rekonstrukci fázového prostoru, je dimenze vnoření  $m_d$ . Tato kapitola vychází z práce [4]. Nejběžnějším způsobem určení  $m_d$  je Metoda nejbližších falešných sousedů. Tzv. sousedé jsou stavy fázového prostoru, které si jsou blízké. Rozlišují se dva druhy sousedů, a to falešní a skuteční. Falešní sousedé si jsou sice v určité projekci blízcí, v jiné jsou už však relativně vzdálení. Např. u trojrozměrného prostoru je možné si pod první projekcí představit pohled v rovině xy a pod druhou pohled v rovině xz. Naproti tomu skuteční sousedé si jsou blízcí ve všech možných projekcích.

Při použití Metody nejbližších falešných sousedů je uvažován  $m_d$ -rozměrný fázový prostor a v něm stav  $X_m(t_i)$ , pro který platí:

$$X_m(t_i) = [x(t_i), x(t_i + \tau), \dots, x(t_i + (m_d - 1) \cdot \tau)] \quad (4.3.3.1)$$

Ke stavu  $X_m(t_i)$  existuje jeho nejbližší soused, stav  $X_m^{NN}(t_i)$ :

$$X_m^{NN}(t_i) = [x^{NN}(t_i), x^{NN}(t_i + \tau), \dots, x^{NN}(t_i + (m_d - 1) \cdot \tau)] \quad (4.3.3.2)$$

Pokud při následném zvýšení dimenze dojde také ke zvětšení vzdálenosti těchto dvou stavů na tolik, že je nebude možné dále považovat za sousedy, jednalo se o falešné sousedy. To, zda je možné stavy  $X_m(t_i)$ ,  $X_m^{NN}(t_i)$  dále považovat za sousedy, určuje poměr  $R(t_i)$ . Aby bylo možné  $R(t_i)$  určit, definují se souřadnice uvažovaných stavů v dimenzi o jedna vyšší:

$$\mathbf{X}_{m+1}(t_i) = [\mathbf{x}(t_i), \mathbf{x}(t_i + \tau), \dots, \mathbf{x}(t_i + (m_d - 1) \cdot \tau), \mathbf{x}(t_i + m_d \cdot \tau)] \quad (4.3.3.3)$$

$$\mathbf{X}_{m+1}^{NN}(t_i) = [\mathbf{x}^{NN}(t_i), \mathbf{x}^{NN}(t_i + \tau), \dots, \mathbf{x}^{NN}(t_i + (m_d - 1) \cdot \tau), \mathbf{x}^{NN}(t_i + m_d \cdot \tau)] \quad (4.3.3.4)$$

Dále se vyjadřuje druhá mocnina vzdálenosti stavů  $X_m(t_i)$  a  $X_m^{NN}(t_i)$  ve fázovém prostoru s dimenzí  $m$   $R_m^2(t_i)$ , pro kterou platí:

$$\mathbf{R}_m^2(t_i) = \|\mathbf{X}_m(t_i) - \mathbf{X}_m^{NN}(t_i)\|^2 \quad (4.3.3.5)$$

a v dimenzi  $m_d + 1$  je vyjádřena jako:

$$\mathbf{R}_{m+1}^2(t_i) = \|\mathbf{X}_{m+1}(t_i) - \mathbf{X}_{m+1}^{NN}(t_i)\|^2 \quad (4.3.3.6)$$

Poměr  $R(t_i)$  je pak definován takto:

$$\mathbf{R}(t_i) = \sqrt{\frac{\mathbf{R}_{m+1}^2(t_i) - \mathbf{R}_m^2(t_i)}{\mathbf{R}_m^2(t_i)}} \quad (4.3.3.7)$$

Řešené vztahy se označí za falešné sousedy, pokud vyjde hodnota  $R(t_i)$  vyšší než stanovený prahový poměr  $R_T$ . Z praxe vychází, že hodnota  $R_T$  by se měla ideálně volit mezi 10 a 15. Za dostatečnou dimenzi vnoření  $m$  je pak považována ta, kdy se počet falešných sousedů blíží nule.

Dalším způsobem, jak určit, zda se jedná o falešné sousedy, je sledování změny jejich vzdálenosti při navýšení dimenze z  $m_d$  na  $m_d+1$ .

$$\frac{|\mathbf{x}(t_i + m_d \cdot \tau) - \mathbf{x}^{NN}(t_i + m_d \cdot \tau)|}{\mathbf{R}_A} \geq \mathbf{A}_T \quad (4.3.3.8)$$

kde  $R_A$  určíme z výrazu:

$$R_A^2 = \frac{1}{N} \sum_{i=1}^N [x(t_i) - \bar{x}]^2 \quad (4.3.3.9)$$

kde  $\bar{x}$  je aritmetický průměr hodnot  $x(t_i)$ . Většinou se uvažuje  $A_T \approx 2$ . Platí-li nerovnice (4.3.3.8), jsou stavy  $X_m(t_i)$  a  $X_m^{NN}(t_i)$  falešnými sousedy.

Druhá metoda pro určení dimenze vnoření se nazývá Caova metoda a vychází z Metody nejbližších falešných sousedů. Je definován poměr  $a(t_i, m_d)$ :

$$a(t_i, m_d) = \frac{R_{m+1}(t_i)}{R_m(t_i)} \quad (4.3.3.10)$$

a dále průměr poměrů  $a(t_i, m)$ ,  $E(m_d)$  definovaný jako:

$$E(m_d) = \frac{1}{N - m_d \cdot \tau} \sum_{i=1}^{N - m_d \cdot \tau} a(t_i, m_d) \quad (4.3.3.11)$$

Pro změnu  $E_1(m_d)$  hodnoty veličiny  $E(m_d)$  platí:

$$E_1(m_d) = \frac{E(m_d + 1)}{E(m_d)} \quad (4.3.3.12)$$

Když je dosaženo dostatečně velké dimenze vnoření  $m_d$ , nevznikají falešní sousedé čili všechny stavy, které si jsou blízké v  $m_d$ -rozměrném prostoru si jsou blízké i v  $m_d + 1$  - rozměrném prostoru. Z toho vyplývá, že dostatečného  $m$  je dosaženo v případě, kdy se hodnota veličiny  $E_1(m_d)$  blíží 1.



#### 4.3.4. Rekonstrukce fázového prostoru v programu MATLAB

Informace v této kapitole vycházejí z centra nápovědy [2]. V programu MATLAB je možné rekonstrukci fázového prostoru provést příkazem `phaseSpaceReconstruction`. Vstupní hodnoty pro tento příkaz se zadávají ve formě časové řady. Mohou být buď ve tvaru vektoru či matice. V případě matice pak program uvažuje každý sloupec za nezávislou časovou řadu. Předem je možné také definovat časové zpoždění a dimenzi vnoření. Obě hodnoty se zadávají jako skalární hodnoty, v případě časových řad ve tvaru matice se časové zpoždění a matice vnoření zadávají ve tvaru vektoru. Tyto vektory musí mít stejnou délku jako je počet řádků matice časových řad.

V případě, že není dimenze vnoření zadána, určí ji program metodou Nejbližších falešných sousedů. Jako dimenzi vnoření pak zvolí nejnižší dimenzi, pro kterou platí, že poměr množství falešných sousedů a všech bodů rekonstruovaného fázového prostoru je menší než vybraná hodnota. Jako výchozí hodnotu program uvažuje 0,1. Pokud není předem definováno časové zpoždění, je jeho hodnota programem určena metodou Míry vzájemných informací I. Jako hodnotu časového zpoždění pak zvolí první lokální minimum míry vzájemných informací.

#### 4.4. Neuronové sítě

Umělá neuronová síť (dále pouze ANN podle Artificial Neural Network) je výpočetní model určený ke zpracovávání informací. Princip, na kterém pracují, je podobný fungování nervové sítě lidského mozku. Buňky nervové soustavy se nazývají neurony. Vzájemně jsou propojeny synaptickými spoji a tvoří tak jakousi síť. Jejich hlavní funkcí je v jednoduchosti přijímat podněty, na jejich základě vytvářet impulzy a ty posílat dalším neuronům. Neuron se skládá z více částí, kde každá plní svoji odlišnou funkci ve dříve zmíněném procesu. První částí neuronu jsou dendrity. Jedná se o vláknovité útvary, které buňku spojují s jinými neurony a přijímají od nich signál. Další částí neuronu je samotné tělo buňky. Jeho účelem je, mimo normálních funkcí buňky, reagovat na příchozí vzruchy tvorbou nervového impulsu. Poslední částí je axon. Ten má tvar silného vlákna a jeho funkcí je šířit nervový impuls vyvolaný v těle buňky do dendritů ostatních neuronů. Výsledný nervový impuls je nejen ovlivněn samotnými příchozími signály, ale také silou synaptického spoje mezi danými neurony. Tyto poznatky vycházejí ze zdroje [3].

Obdobným způsobem jsou pak sestaveny ANN. Neurony nervové soustavy jsou zastoupeny jednoduchými procesory, které jsou označovány jako uzly, anebo též neurony, obdobně jako jejich organická předloha. Tyto uzly jsou navzájem propojeny váženými spoji, ekvivalenty dendritů. To, jakou má spoj váhu, určuje, jak velký vliv má signál vysílaný prvním neuronem na výstup druhého neuronu. Hodnota váhy se může pro každý spoj lišit a může dosahovat jak kladných, tak záporných hodnot. Přijímání signálu přes vážené spoje je podle [6] první z hlavních funkcí uzlu. Další funkcí je zpracování přijatých vážených signálů a sestavení nového výstupního signálu. Třetí a poslední je aktivační funkce, která má dvě úlohy, a to omezování hodnot výstupního signálu a vnesení nelinearity do systému. Signál je poté zaslán dalším neuronům a ty ho opět s různou vahou přijímají.

Zpracování vstupních signálů podle [7] probíhá tak, že jsou nejprve sečteny hodnoty vážených signálů a k nim je přičten práh:

$$y = \sum (w_i \cdot x_i) + b$$

kde  $y$  je výsledný signál,  $b$  je práh,  $x_i$  jsou hodnoty jednotlivých vstupních signálů a  $w_i$  jsou k nim příslušné váhy.

Výsledný signál následně projde aktivační funkcí a tím je sestaven výstupní signál. Práh b slouží k posunutí lineární funkce doleva nebo doprava, čímž dodává uzlu výraznou flexibilitu ve způsobu, jakým ze vstupních dat sestavuje výstup. Hodnota prahu je pro každý neuron odlišná a není závislá na žádné jiné hodnotě. Aktivační funkce má za úkol usměrnit výstupní signál takovým způsobem, aby se jeho hodnota pohybovala v rámci stanovených mezí. Používají se různé meze, ale nejčastěji se hodnoty výstupu omezují buď mezi 0 a 1, nebo -1 a 1. Druhou úlohou aktivační funkce je vnést do systému nelinearitu, čímž výrazně urychlí zpracování některých problémů a umožní vyřešit takové, u kterých by to jinak nebylo ani možné. Existuje mnoho druhů aktivačních funkcí, např. funkce identity, sigmoidní funkce, hyperbolický tangens a ReLU funkce.

Uzly jsou podle [6] v rámci ANN sestaveny do vrstev. Tyto vrstvy se dělí do tří kategorií. První je vstupní vrstva, do které jsou dodávány vstupní data řešeného problému. Dalším typem je výstupní vrstva, která sestavuje výsledné výstupy řešeného problému. Mezi těmito vrstvami se nacházející skryté vrstvy, které postupně mění hodnoty signálu ze vstupních na výstupní. Neuronová síť vždy obsahuje alespoň jednu skrytou a jednu výstupní vrstvu, ale skutečný počet závisí až na konkrétní aplikaci sítě. Stejně tak počet neuronů ve vrstvách se pro každý konkrétní model liší, většinou ani vrstvy jedné sítě neobsahují stejný počet uzlů. Avšak každý neuron by měl být připojen ke všem neuronům z předchozí vrstvy.

#### **4.4.1. Učení ANN**

Poté co je neuronová síť sestavena, je nutné, aby prošla procesem učení. Protože stejně jako lidský mozek vyžaduje úpravy synaptických spojů mezi neurony, potřebuje i ANN nastavit své parametry, aby dokázala data zpracovávat správným způsobem. Těmito parametry jsou váhy jednotlivých spojů a prahy jednotlivých neuronů.

Podle [7] existují dva základní druhy učení ANN. První typ se nazývá učení s dohledem. V tomto případě je učení prováděno tak, že jsou síti dodány vstupy, které zpracuje a výsledky jsou porovnány s také dodanými výstupy. Rozdíly získané z tohoto srovnání se nazývají chyby. Ty jsou zpět dodány systému, který následně provede korekci parametrů, opět zpracuje vstupní data, porovná nové výsledky a získá nové hodnoty chyb. Tento proces se opakuje, dokud nejsou rozdíly mezi danými a vypočtenými výstupy přijatelně malé. Tento způsob se nazývá Algoritmus zpětného šíření chyby. Naopak při

učení bez dohledu nejsou správné výstupy dostupné a systém je tak nucen se učit metodou pokusu a omylu. Systém se v tomto případě snaží vytvořit vhodnou zkomprimovanou reprezentaci vstupních dat. To je prováděno snížením dimenzionality, vyhlazením, nebo prosté redukci vstupních dat na konečný počet. Tyto systémy se pak využívají pro klasifikaci dat a hledání anomálií nebo skrytých vlastností ve vstupních datech.

Při učení může podle [6] dojít ke dvěma negativním efektům. První z nich se nazývá overfitting neboli přehlcení. Dochází k němu, když je neuronová síť pro daný problém příliš komplikovaná, když obsahuje příliš mnoho vrstev, nebo i neuronů. V tomto případě se síť příliš přizpůsobila tréninkovým datům, a pokud jí jsou podány jiná data, nedokáže je správně zpracovat čili má slabou schopnost je zobecňovat. Druhý problém nazývaný underfitting neboli nedosycení nastává v případě, že je síť pro řešení daných problémů příliš jednoduchá a není schopna z dat získat přijatelný výsledek.

#### **4.4.2. Druhy ANN**

Podle [7] existují dva základní typy ANN. Prvním z nich jsou dopředné neuronové sítě (dále pouze FFNN, podle feedforward neural network). Ve FFNN jsou informace posílány pouze jedním směrem, a to ze vstupních uzlů do uzlů první skryté vrstvy, z nich do uzlů druhé skryté vrstvy atd., až nakonec proudí z poslední skryté vrstvy do výstupních neuronů. Nedochází tudíž k žádné zpětné vazbě. Druhým typem ANN jsou zpětné neuronové sítě (dále pouze FBNN podle feed-backward neural network). V tomto případě obdobně zasílají neurony dané vrstvy výstupní signály neuronům následující vrstvy. Ty jim ale poté zasílají zpětnou vazbu, podle které se neurony dané vrstvy přizpůsobují, a k úpravě výstupů tak dochází už v rámci jednoho výpočtu.

#### 4.4.3. Pozice ANN vůči jiným výpočetním modelům

V této kapitole vycházím ze zdrojů [6] a [7]. ANN již řadu let dokazují, že jsou relativně konkurence schopné vůči běžně užívaným regresním a statistickým modelům a v určitých oblastech je dalece přesahují. Jejich velký potenciál spočívá v tom, že dokážou souběžně velmi rychle vyřešit mnoho zadaných úloh. Mají mnoho užitečných vlastností jako učení sama sebe, přizpůsobovat a opravovat se, pracovat i s nekompletními daty, odolávat šumu a chybám ve vstupních datech, řešit nelineární problémy a mapovat cestu od vstupu k výstupu. Jednou z nejvýznamnějších vlastností ANN je schopnost zobecňovat čili predikovat případy, které nejsou součástí dat použitých při učení sítě. Díky těmto vlastnostem jsou ANN schopné řešit i takové problémy, které není možné vyřešit výpočetními možnostmi tradičních postupů a konvenční matematiky. Systém ANN však silně závisí na dodaných datech pro učení. Pokud jich není dostatečné množství, nepokrývají dostatečnou část podmínek řešeného systému, nebo obsahují výrazný šum, nemusí neuronová síť nikdy dosáhnout požadovaných výsledků. Přes všechny nesporné výhody ANN je však stále mnoho prostoru pro jejich výzkum a další vývoj.

ANN se používají např. pro rozpoznávání řeči, rukopisu, obličejů a různých 3D objektů, obnovení poškozených telekomunikačních dat, výklad více jazykových zpráv, analýzu struktury objektu, hledání ložisek ropy, vyhledávání námořních min, vývoje akciového trhu, předpověď počasí a klimatických změn, předpovědi velikosti zemědělské úrody, aproximace funkcí v numerických modelech.

#### 4.4.4. Neuronové sítě v MATLAB

V této kapitole jsem vycházel z informací nacházejících se v centru nápovědy jako je [2] a další. V programu MATLAB je možné vytvořit mnoho různých typů ANN. Základní dva druhy jsou mělké neuronové sítě a sítě hlubokého učení. Mělké NN se skládají z jedné výstupní vrstvy a jedné nebo několika skrytých vrstev. Neobsahují vstupní vrstvu v pravém slova smyslu, pouze vrstvu vstupů, které jsou zasílány přímo do první skryté vrstvy. Pro případ, kdyby chtěl uživatel vytvořit složitější ANN, použije hluboké učení. V tomto případě může uživatel určit například množství vrstev, aktivační funkce neuronů, zpětnou vazbu mezi vrstvami a mnoho dalších parametrů a vlastností pro tvorbu svojí ANN. Tento způsob však vyžaduje vysoký stupeň porozumění problematice, kterého lze dosáhnout pouze dlouhým studiem tohoto oboru a dlouhodobou praxí. Naproti tomu tvorba mělkých NN je značně jednodušší. V tomto programu je k dispozici několik typů mělkých NN. Jednou z nich je

běžná dopředná NN, kde informace proudí pouze jedním směrem a pouze k neuronům další vrstvy. Tato NN se vytváří příkazem `feedforwardnet`. Dalším typem je kaskádová dopředná síť, kde informace proudí nejen do další vrstvy, ale i do všech, které následují za ní. Pro tuto síť platí příkaz `cascaforwardnet`. Existuje zde také nelineární autoregresivní NN pod příkazem `narxnet` a síť pro rozpoznávání vzorců `patternnet`.

Při tvorbě dopředné neuronové sítě pomocí `feedforwardnet` může uživatel zvolit počet skrytých vrstev, množství jejich neuronů a algoritmus, který bude použit pro učení sítě. Na výběr je nemalé množství algoritmů učení. V programu každému náleží příslušná funkce viz Tabulka 2. Výchozí funkcí programu je `trainlm`, ve většině případů umožňuje nejrychlejší proces ze všech algoritmů a je proto doporučován jako první volba při tvorbě nové ANN.

Funkce	Algoritmus
<code>trainlm</code>	Levenberg-Marquardtův
<code>trainbr</code>	Bayesianova regularizace
<code>trainbfg</code>	BFGS Kvazi-Newtonův
<code>trainrp</code>	Odolné zpětné šíření
<code>trainscg</code>	Stupňovaný sdružený gradient
<code>traincgb</code>	Sdružený gradient s Powell/Bealeovým restartem
<code>traincgf</code>	Fletcher-Powellův sdružený gradient
<code>traincgp</code>	Polak-Ribièreův sdružený gradient
<code>trainoss</code>	Jednokrokové sečny
<code>traingdx</code>	Gradientní sestup s variabilní rychlostí učení
<code>traingdm</code>	Gradientní sestup s hybností
<code>traingd</code>	Gradientní sestup

*Tabulka 2: Dostupné algoritmy pro učení neuronových sítí v programu MATLAB, převzato a upraveno z [2]*

## 5. Vlastní práce – rychlost větru

V této kapitole jsou popsány postupy vypracování úkolů diplomové práce týkajících se rychlosti větru.

### 5.1. Zpracování a analýza dat rychlosti větru

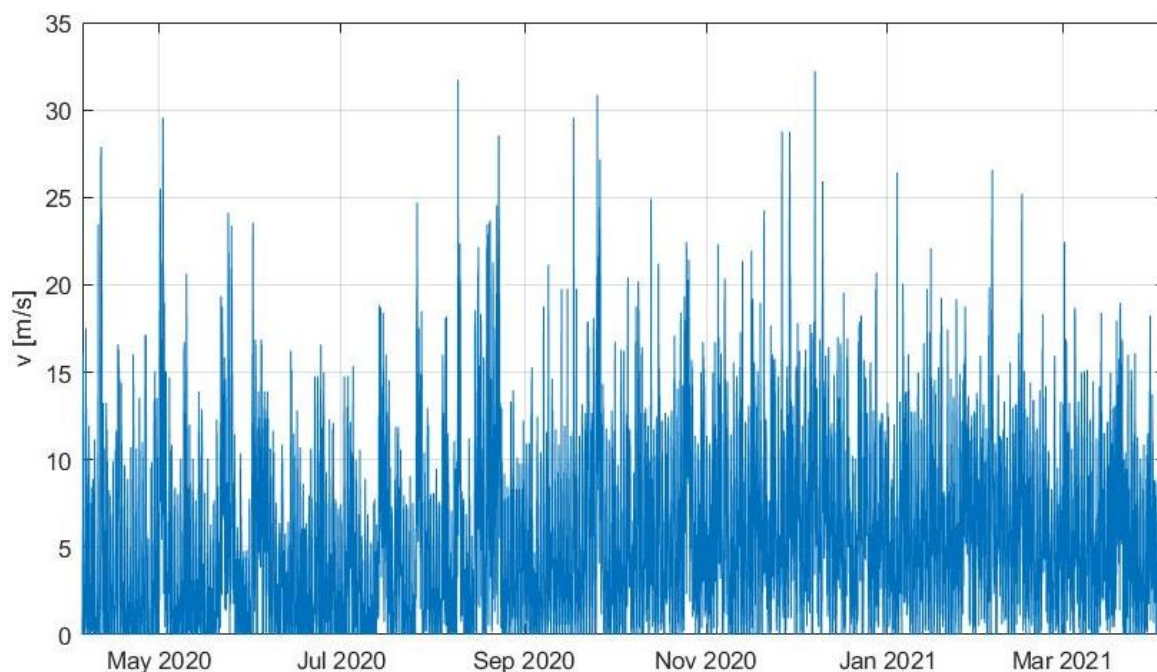
První veličina, která byla v rámci této práce zkoumána, byla rychlost větru. Data rychlosti větru byla získána z portálu Thingspeak (Thingspeak.com), což je podle [2] analytická platforma umožňující shromažďovat, vizualizovat, zpracovávat a analyzovat různorodá data. Hodnoty byly naměřeny soukromou meteorologickou stanicí nacházející se v Sydney (Nový Jižní Wales, Austrálie). Stanice měří hodnoty každou minutu a ihned je odesílá na kanál na portálu Thingspeak. Tam je k datům majitelem stanice umožněn přístup veřejnosti a jejich použití mimo jiné i pro akademickou činnost. Z cloudového úložiště webu byla data rychlosti větru získána přímo do prostředí programu MATLAB pomocí příkazu `thingSpeakRead`. Ten umožňuje uživateli stáhnout data z vybraného kanálu webu, na kterém jsou uložena viz. Obrázek 3. V rámci této funkce si může uživatel zvolit, která data a z jakého časového úseku stáhne. Také k nim může získat příslušné časy a datumy, kdy byly jednotlivé hodnoty naměřeny. Nedostatkem této funkce je, že umožňuje stáhnout maximálně 8000 hodnot. Z toho důvodu je třeba ji v případě potřeby většího množství dat použít několikrát.

```
[dataA, timeA] = thingSpeakRead(202058, 'Fields', 2, 'DateRange', [datetime  
(2020, 10, 12, 00, 00, 01), datetime(2020, 10, 17, 00, 00, 01)]);
```

Obrázek 3: Příklad použití funkce pro nahrávání dat z portálu Thingspeak

Následně byla na datech rychlosti větru provedena analýza z hlediska potenciálu produkce elektrické energie. Jak už dříve v práci zmíněno, čím vyšší rychlostí fouká vítr, tím větší množství energie z něj lze teoreticky získat. Ve skutečnosti tato úměra neplatí bez omezení. Turbíny jsou konstruovány na daný výkon, kterého dosáhnou při určité rychlosti větru, a další zvyšování již vyšší produkce elektrické energie neumožní. Při relativně příliš velkých rychlostech je dokonce nutné, aby byly větrné elektrárny zastaveny, aby se vyvarovaly možnému poškození. Analýza rychlosti větru byla provedena pomocí Weibullova grafu. Z webu byla získána data naměřená za jeden kalendářní rok a následně z nich byly vypočítány aritmetické průměry po desetiminutových intervalech. Tímto

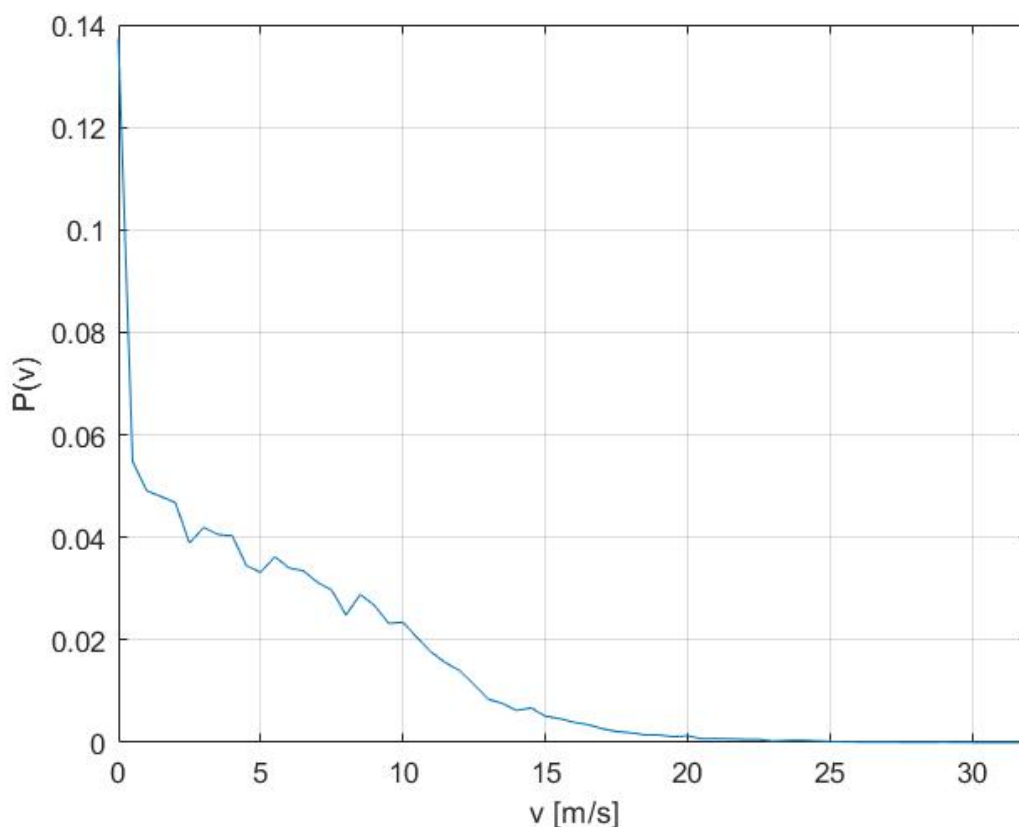
krokem byl výrazně snížen vliv prudkých změn rychlostí větru na výstup analýzy. Tyto prudké změny jsou totiž způsobovány poryvy a turbulencemi a mohou ovlivňovat naměřené hodnoty. Přitom však mají pouze minimální vliv na výkon větrné elektrárny kvůli setrvačnosti rotoru. Ve větrném průmyslu je navíc běžnou praxí pracovat s průměrnými hodnotami rychlosti větru za deset minut. Následně byly hodnoty zaokrouhleny na nejbližší násobek pěti desetin, aby odpovídaly hodnotám výkonové křivky, kterou byla později na data aplikována. Z takto upravených dat byl sestaven graf viz. Obrázek 4.



Obrázek 4: Data naměřená za jeden kalendářní rok získaná z portálu z Thingspeak

Rychlostem větru, kterých bylo podle naměřených dat dosaženo, byla následně podle četnosti výskytu přiřazena hodnota pravděpodobnosti, která byla zapsána do Weibullova grafu, viz Obrázek 5. Z grafu je zřejmé, že zvolené místo není pro umístění větrné elektrárny ideální. Relativně velkou část času zde totiž vítr nefouká, nebo vane velmi malou rychlostí. S rostoucí rychlostí klesá pravděpodobnost, že jí bude vítr foukat a u hodnot větších než 13 m/s je již dokonce menší než 0,01. Byla také průměrnou rychlost větru za celý rok, která činila přibližně 5,2 m/s. Toto je však pouze orientační údaj, pro posouzení dat z pohledu produkce elektrické energie nemá velkou výpovědní hodnotu.





Obrázek 5: Weibullův graf pro rychlost větru ve výšce 10 m rozdělenou po 0,5 m/s

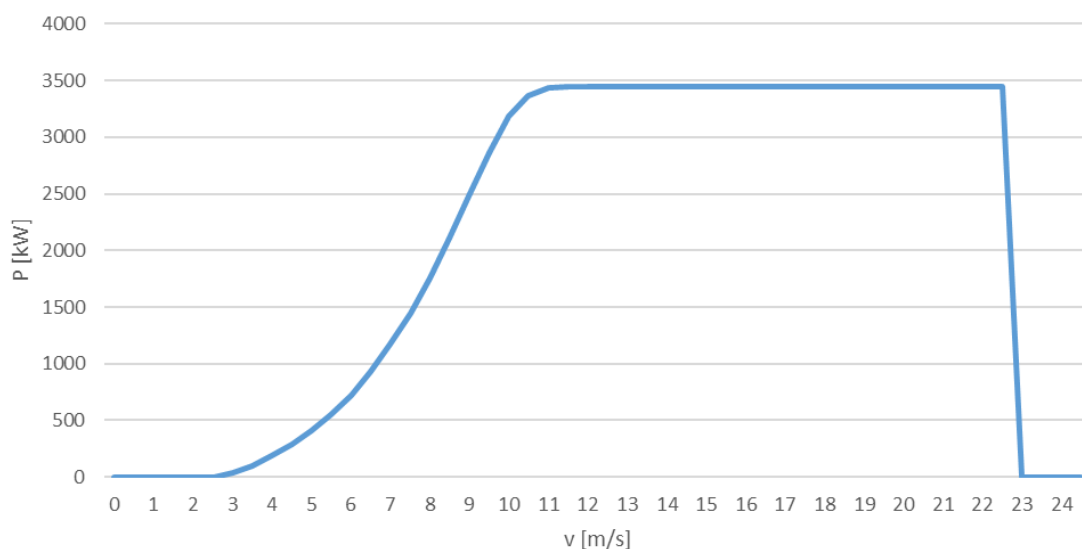
## 5.2. Výkon větrné elektrárny

Pro výpočty potenciačního výkonu větrné elektrárny umístěné v místě měření byla vybrána turbína V126 - 3,45 MW od firmy Vestas. Je to jedna z nejběžněji používaných turbín v Novém Jižním Walesu a je vhodná na středně a mírně větrná stanoviště. Na webových stránkách výrobce [11] bylo uvedeno, že tato turbína se běžně umísťuje do výšek 87, 117, 137 nebo 147 v závislosti na rychlosti větru. Výrobce také uvádí, že turbína začíná produkovat elektrickou energii při rychlosti větru 3 m/s a je třeba ji vypnout při rychlostech vyšších než 22,5 m/s. Celá výkonová křivku byla získána ze zdroje [13] viz. Tabulka 3 a Obrázek 6.

v [m/s]	P [kW]	v [m/s]	P [kW]
0	0	12,5	3450
0,5	0	13	3450
1	0	13,5	3450
1,5	0	14	3450
2	0	14,5	3450
2,5	0	15	3450
3	35	15,5	3450
3,5	101	16	3450
4	184	16,5	3450
4,5	283	17	3450
5	404	17,5	3450
5,5	550	18	3450
6	725	18,5	3450
6,5	932	19	3450
7	1172	19,5	3450
7,5	1446	20	3450
8	1760	20,5	3450
8,5	2104	21	3450
9	2482	21,5	3450
9,5	2865	22	3450
10	3187	22,5	3450
10,5	3366	23	0
11	3433	23,5	0
11,5	3448	24	0
12	3450	24,5	0

Tabulka 3: Data pro výkonovou křivku turbíny V126 - 3,45 MW Vestas

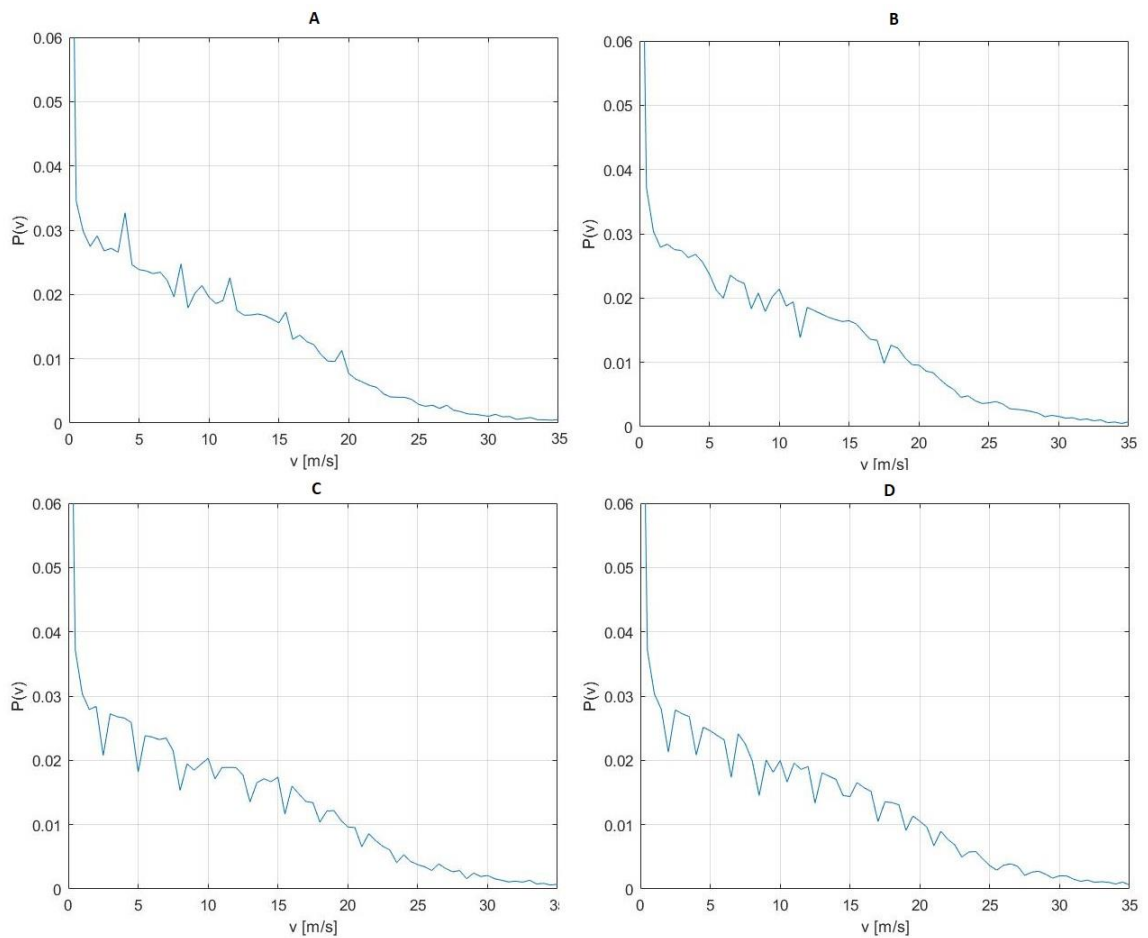
Výběr výšky umístění turbíny byl proveden porovnáním průměrných výkonů, kterých by turbína dosahovala ve všech běžně používaných výškách. K takovému výpočtu však bylo nejprve nutno zjistit, v jaké výšce bylo prováděno měření. Na webu thingspeak není uvedeno, v jaké poloze je anemometr meteorologické stanice umístěn. Proto bylo předpokládáno, že se nacházel ve výšce 10 metrů, jako je v praxi pro měření rychlosti větru běžné. Dále bylo nutné zjistit třídu drsnosti krajiny, v níž byly data zaznamenány. Na kanálu jsou uvedeny souřadnice, kde se meteorologická stanice nachází. Podle nich bylo zjištěno, že měření byly prováděny na předměstí Sydney, kde se v okolí nachází velké množství nízkých budov a stromů. Podle Tabulka 1 byla tomuto prostředí přiřazena třída drsnosti 3 což odpovídá délce drsnosti 0,4 m. Následně byly podle vzorce (4.1.1.1) přepočítány data rychlosti větru do všech čtyřech užívaných výšek. Bylo předpokládáno, že čím výše se turbína větrné elektrárny nachází, tím více elektrické energie může vytvářet. S rostoucí výškou však rostou i náklady na konstrukci a výstavbu elektrárny, proto není výhodné se za každou cenu snažit zvýšit potenciální výkon, ale je třeba hledat jejich nejvýhodnější poměr.



Obrázek 6: Výkonová křivka turbíny V126 – 3,45 MW Vestas

Z přepočítaných hodnot byly následně sestaveny Weibullové grafy pro všechny možné výšky umístění turbíny, viz. Obrázek 7. Následně byly z grafů a výkonové křivky turbíny vypočteny průměrné výkony, kterých by bylo v možných výškách umístění dosahováno, viz. Tabulka 4.

Vyobrazené grafy jsou zaměřené na část Weibullova rozdělení, kde se nacházejí ty hodnoty rychlosti větru, při kterých vybraná turbína pracuje. Je možné pozorovat, že s rostoucí výškou umístění turbíny roste pravděpodobnost výskytu vyšších rychlostí větru. Z Tabulka 4 je zřejmé, jak velký vliv má pozice turbíny na její výkon. Přestože oblast, kde byly měření prováděny, není pro větrnou elektrárnu ideální, umístění turbíny do výšky 87 metrů umožní téměř poloviční nárůst oproti výkonu vypočtenému z naměřených dat čili průměrně 1502,2 kW. Ve výšce 117 metrů by turbína dosahovalo průměrného výkonu 1521,16 kW. Může se zdát, že jde o pouze relativně malý nárůst, ale za celou dobu životnosti turbíny se jedná o poměrně významný rozdíl. Ve výšce 137 resp. 147 metrů pak průměrný výkon vychází jako 1525,37 resp. 1525,59 kW, což už jsou opravdu minimální rozdíly, které by pravděpodobně dostatečně nekompenzovaly nárůst nákladů na výstavbu. Proto bylo rozhodnuto umístit teoretickou turbínu do výšky 117 metrů, kde bude dosahovat průměrného výkonu 1521,16 kW, což je pouze 44 % maximálního výkonu této turbíny.



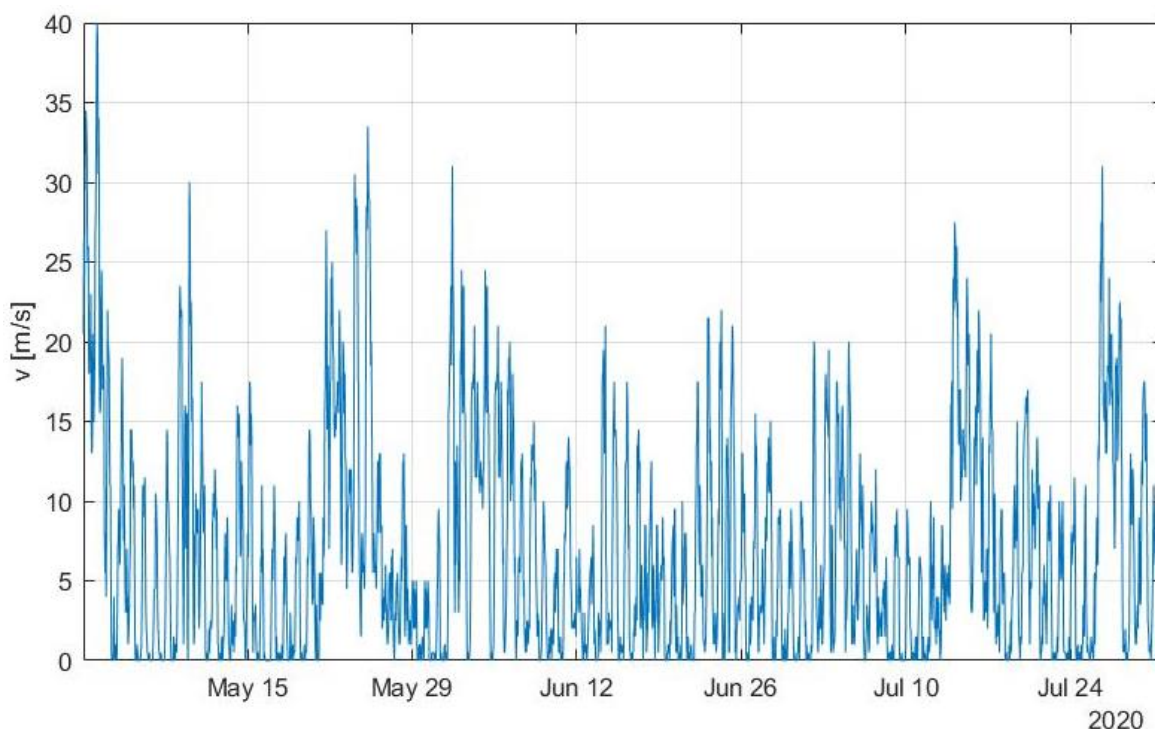
Obrázek 7: Weibullové grafy, A - ve výšce 87 m, B - ve výšce 117 m, C - ve výšce 137 m, D - ve výšce 147 m

h [m]	P [kW]
10	986,49
87	1502,20
117	1521,16
137	1525,37
147	1525,59

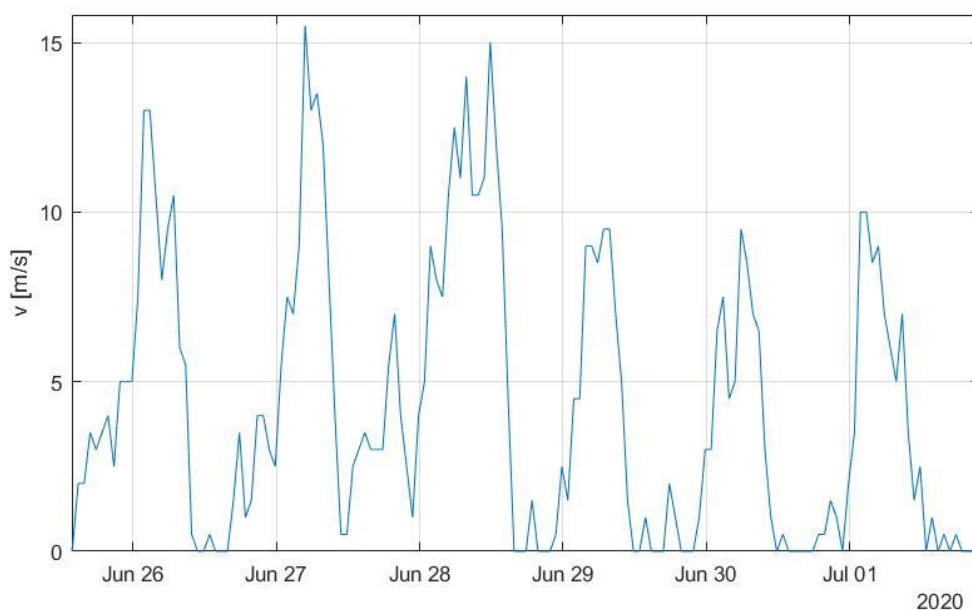
Tabulka 4: Průměrný výkon větrné elektrárny

### 5.3. Analýza časové řady pro učení neuronové sítě

Predikce rychlosti větru byla prováděna v programu MATLAB. Pro učení neuronové sítě byla vybrána data naměřená během května, června a července 2020. Jednalo se totiž o jedno z nejdelších období, během kterého při měření nenastaly výrazné výpadky. I v těchto datech však několikrát chyběly hodnoty pro několik minut, v takovém případě byly zanedbány a průměrná rychlost větru se určila pouze z dostupných hodnot. Nakonec bylo rozhodnuto vypočítat průměrné hodnoty za hodinový interval. Účelem volby tohoto delšího intervalu bylo potlačit vliv nárazových změn rychlosti větru a případných chyb měření. Díky tomu byl také potlačen šum a byl více patrný cyklus časové řady. Výsledné průměrné rychlosti byly stejným způsobem jako v předchozí kapitole přepočítány na hodnoty ve výšce 117 metrů viz Obrázek 8. Z tohoto grafu, a zvláště pak z jeho detailu viz Obrázek 9 lze pozorovat každodenní cyklus časové řady. Na začátku dne rychlost stoupá, maxim dosahuje v noci a v brzkých ranních hodinách. Během dopoledne rychlost klesá většinou až k nulové hodnotě, blízko které se drží celé odpoledne a večer a s počátkem noci začíná rychlost větru opět narůstat. Takovýto cyklus je poměrně atypický, je totiž běžně že vítr dosahují největších rychlostí během dne, kdy sluneční energie způsobuje rozdílné teploty vzduchu nad zemí a oceánem, což způsobuje jeho pohyb ve formě větru.



Obrázek 8: Data rychlosti větru použítá pro učení neuronových sítí



Obrázek 9: Detail grafu na Obrázek 8 (datum na vodorovné ose značí začátek dne)

Takto upravená chaotická časová řada byla následně v programu MATLAB analyzována příkazem `phaseSpaceReconstruction` viz Obrázek 10 a byl z ní rekonstruován fázový prostor. Ideální hodnoty dimenze vnoření a časového zpoždění fázového prostoru program vypočítal jako  $m_d = 5$  a  $\tau = 8$ . Fázový prostor tedy měl tvar matice o pěti řádcích a každá hodnota v  $i$ -tém sloupci se rovnala hodnotě v  $i+8$ -mém sloupci předchozího řádku.

```
[XR, lag, dim] = phaseSpaceReconstruction(data);
```

Obrázek 10: Příkaz pro rekonstrukci fázového prostoru, kde *data* je časová řada, *XR* je fázový prostor, *lag* je časové zpoždění fázového prostoru ( $\tau$ ) a *dim* je počet dimenzí fázového prostoru ( $m_d$ )

#### 5.4. Proces učení neuronové sítě

Dalším krokem bylo vytvořit dopřednou neuronovou síť. Jejím vstupem byl rekonstruovaný fázový prostor, který měl tvar matice a každý sloupec hodnot tak představoval jednu sadu vstupů. Předem daným výstupem pak byla původní časová řada, z níž byl fázový prostor sestaven. Jednu sadu výstupních dat tak tvořila pouze jedna hodnota. Časová řada byla vůči prvnímu řádku fázového prostoru o jednu pozici posunuta. Tímto způsobem bylo dosaženo toho, že se neuronová síť učila predikovat rychlost větru v čase  $i$  na základě hodnot v časech  $i - 1$ ,  $i - (\tau + 1)$ ,  $i - (2 \cdot \tau + 1)$ , ...,  $i - ((m_d - 1) \cdot \tau + 1)$ . Neuronová síť tedy měla pět vstupů, jeden neuron ve výstupní vrstvě a jeden výstup. K vytvoření neuronové sítě byl použit příkaz `feedforwardnet`, viz Obrázek 11. Ve výchozím stavu je tímto příkazem vytvořena jednoduchá síť obsahující jednu skrytou vrstvu o deseti neuronech a jednu výstupní vrstvu. Počet skrytých vrstev a jejich neuronů však může být změněn uživatelem. Uživatel také může předem určit, podle jakého algoritmu budou parametry (prahy a váhy) neuronové sítě sestaveny. Pro učení je v programu MATLAB k dispozici několik algoritmů. Na doporučení vedoucího práce z nich byly vybrány dva, které se pro podobné problémy používají nejčastěji. Těmito algoritmy byly Levenberg-Marquardtův (dále pouze LM) algoritmus a Bayesianova regularizace (dále BR). Oba tyto algoritmy učí neuronové sítě na principu zpětného šíření chyby, kdy je pro každou sadu parametrů vypočítán jakobián výkonnosti sítě a na jeho základě jsou parametry dále upravovány. Tento jakobián je vypočten z rozdílů dodaných výstupů a výstupů vypočítaných neuronovou sítí.

```
net = feedforwardnet(hiddenSizes, trainFcn);
```

*Obrázek 11: Příkaz pro vytvoření dopředné neuronové sítě, kde `hiddenSizes` je vektor udávající počet skrytých vrstev a množství jejich neuronů, `trainFcn` udává, jako algoritmus bude použit při učení sítě a `net` je vytvořená neuronová síť*

Výběr, který z těchto algoritmů měl být pro učení sítě použit, byl proveden na základě přesnosti výsledků neuronových sítí. Stejně tak byl proveden i výběr počtu skrytých vrstev sítě a množství jejich neuronů. Do porovnání byly zařazeny sítě s jednou skrytou vrstvou, které měly 1 až 30 neuronů, a dvěma skrytými vrstvami, kde vždy obě měly stejný počet neuronů a to 1 až 15. Učení každého z těchto typů sítí proběhlo podle obou algoritmů. Každá z těchto variant sítí podstoupila učení dvacetkrát, aby bylo zajištěno, že bude dosaženo co nejlepších výsledků. Ty byly vyhodnocovány pomocí funkce `mse` viz Obrázek 13, která v programu MATLAB slouží k hodnocení výkonnosti neuronových sítí. Vstupy pro tuto funkci tvoří neuronová síť, správné hodnoty výstupů a vypočítané hodnoty

výstupů. Funkce spočítá střední kvadratickou odchylku vypočítaných výstupů vůči správným. Výsledky tohoto průzkumu byly zpracovány do Tabulka 5 **Chyba! Nenalezen zdroj odkazů.** Zde je možné pozorovat, že s rostoucím množstvím neuronů se zlepšuje schopnost neuronové sítě predikovat správné výsledky. Též z ní vyplývá, že pokud je použit LM algoritmus, dosahují při stejném celkovém počtu neuronů lepších výsledků sítě s jednou vrstvou. Je-li zvolen algoritmus BR platí to při menším počtu neuronů také, ale s jejich rostoucím množstvím se rozdíly zmenšují, a pro 14 a více neuronů už dosahují lepších výsledků neuronové sítě se dvěma a více vrstvami. Ve srovnání algoritmů dosahují lepších výsledků sítě učené podle BR algoritmu. Získání lepších výsledků však má svoji cenu, s rostoucím počtem neuronů také výrazně roste počet parametrů, které je třeba optimalizovat, a tím také roste výpočetní náročnost procesu učení. Zvláště při učení pomocí BR algoritmu vyžaduje trénink více času. Nejlepších výsledků dosahovala neuronová síť se dvěma skrytými vrstvami o 15 neuronech, která byla učená BR algoritmem. Proto byla vybrána pro použití při predikci. Pomocí příkazu `getwb`, viz Obrázek 13, byly získány všechny hodnoty vah a prahů vybrané neuronové sítě.

```
k = 20;
NN = cell(1, k);
perfs = zeros(1, k);

for i = 1:k
    NN{i} = train(net, vstupy, cile);
    vystup = NN{i}(vstupy);
    perfs(i) = mse(net, cile, vystup);
end
```

Obrázek 12: Část skriptu určená pro trénink a hodnocení výkonnosti neuronových sítí, kde `net` a `NN` jsou neuronové sítě, `vstupy` jsou vstupy neuronových sítí, `cile` jsou dodané výstupy, `vystup` jsou výstupy určené sítí a `perfs` jsou hodnocení výkonnosti sítí



Počet neuronů	$\sigma^2(v)$ [m <sup>2</sup> /s <sup>2</sup> ]			
	LM algoritmus		BR algoritmus	
	jedna skrytá vrstva	dvě skryté vrstvy	jedna skrytá vrstva	dvě skryté vrstvy
1	6.85	6.85	6.84	6.84
2	6.68	6.70	6.67	6.61
3	6.58	6.56	6.55	6.38
4	6.50	6.45	6.44	6.18
5	6.48	6.35	6.33	6.00
6	6.40	6.42	6.24	5.67
7	6.39	6.33	6.15	5.29
8	6.34	5.94	6.06	5.13
9	6.39	6.21	5.94	5.04
10	6.15	5.94	5.90	4.76
11	6.34	6.21	5.68	4.55
12	6.38	6.82	5.60	4.47
13	6.20	5.78	5.56	4.23
14	6.10	5.82	5.42	4.21
15	6.10	5.89	5.31	4.05
16	6.00		5.22	
17	6.08		5.09	
18	5.95		5.08	
19	6.00		5.13	
20	5.97		4.95	
21	5.98		4.92	
22	5.97		4.89	
23	5.95		4.82	
24	5.95		4.67	
25	5.88		4.76	
26	5.82		4.52	
27	5.74		4.57	
28	5.80		4.57	
29	5.69		4.46	
30	5.66		4.54	

Tabulka 5: Srovnání přesnosti predikce různých sítí a algoritmů učení pomocí směrodatné kvadratické odchylky

```
parametry = getwb(net);
```

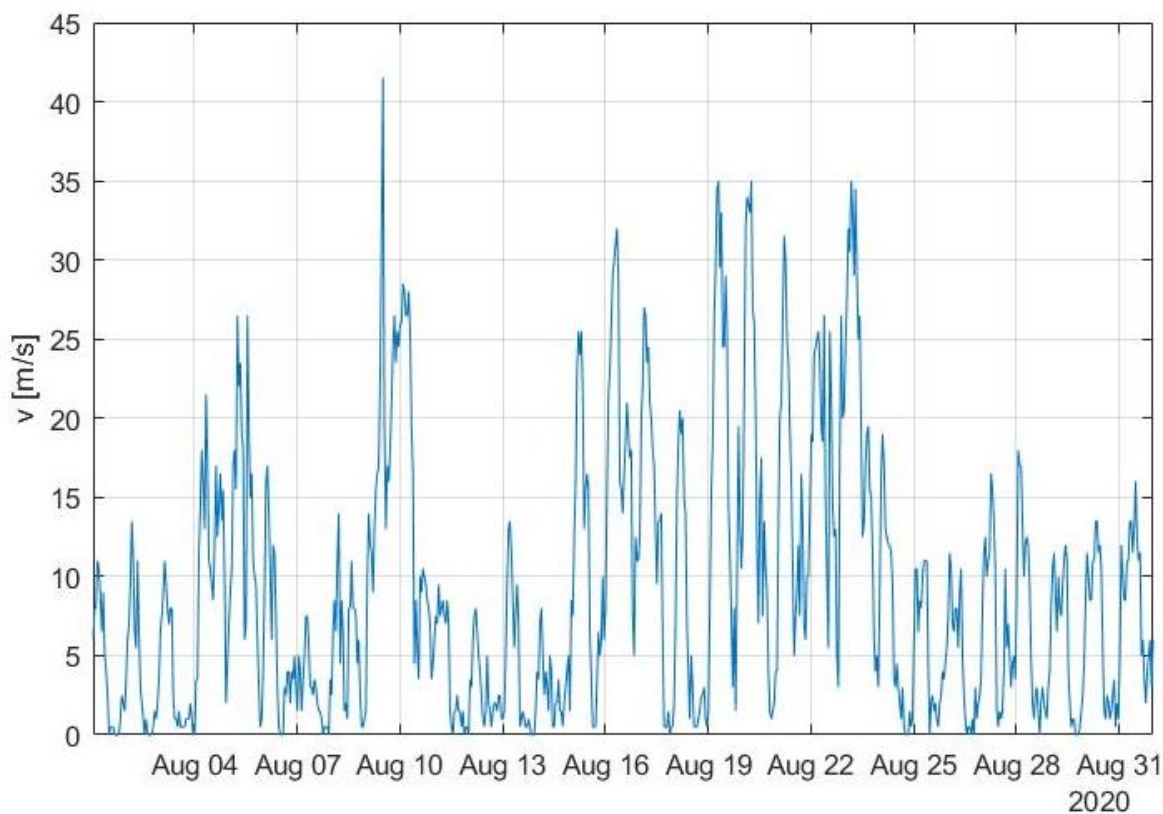
Obrázek 13: Příkaz pro získání parametrů neuronové sítě, kde net je neuronová síť a parametry jsou její váhy a prahy

## 5.5. Predikce

Aby podmínky předpovědi odpovídaly skutečnosti, bylo třeba ji provádět na odlišných datech, než na kterých byla neuronová síť učena. Proto byla pro predikci použita data naměřená v srpnu 2020. Aby byla neuronová síť schopná pracovat správným způsobem, musely se hodnoty parametrů  $\tau$  a  $m_d$  tohoto fázového prostoru rovnat hodnotám parametrů prostoru použitého pro učení neuronové sítě, tedy  $\tau = 8$  a  $m_d = 5$ . Při učení neuronová síť predikovala pouze následující hodnotu rychlosti větru, při předpovědi je však třeba dokázat určit hodnoty rychlosti větru alespoň několik kroků dopředu. Aby tohoto bylo docíleno, musel být skript sestavit takovým způsobem, aby se vypočítaná výstupní data stala novými vstupními daty pro predikci následující hodnoty. To bylo provedeno způsobem zobrazeným na Obrázek 15. Predikované hodnoty byly vzápětí po každém výpočtu umístěny na všechny příslušné pozice. Na počátku predikce bylo třeba znát hodnoty rychlosti větru v alespoň  $k$  předchozích časech, kde:

$$k = \tau \cdot (m_d - 1), \quad (5.3.1)$$

aby z nich mohl být rekonstruován fázový prostor sloužící jako vstupní data predikce. Potom platilo, že  $k = 32$ .



Obrázek 14: Data rychlosti větru použitá pro predikci pomocí neuronové sítě

```

k = lag*(dim-1);
vystup = cile(1:k);

for i = k:(N+k)
    vystup(i) = net(vstupy(:,i));
    for r = 1:dim
        vstupy(dim+1-r,i+1) = vystup(i-(r-1)*lag);
    end
    r = 1;
end

```

Obrázek 15: Část skriptu určená pro predikci na  $N$  kroků dopředu, kde  $lag$  je časové zpoždění fázového prostoru  $\tau$  a  $dim$  je počet jeho dimenzí  $m_d$ ,  $cile$  jsou dodaná výstupní data,  $vystup$  jsou vypočítaná výstupní data a  $vstupy$  jsou vstupní data

Následně byla sestavena vybraná neuronová síť. Pomocí příkazu `feedforwardnet` byla vytvořena dopředná neuronová síť se dvěma skrytými vrstvami po patnácti neuronech. Funkcí `configure` byl podle vstupních a výstupních dat nastaven její počet vstupů a výstupů. Příkazem `setwb` byly dosazeny prahy všech neuronů a váhy jejich spojů, které byly určeny při procesu učení a uloženy viz Obrázek 16. Následně už mohla být provedena samotná predikce. Aby mohlo být objektivně posouzeno, jak byla predikce přesná, byla provedena několikrát s počátkem v různých bodech. Výsledky jsou uvedeny v kapitole 7 této práce.

```

net = feedforwardnet([15,15]);
net = configure(net, vstupy, cile);
net = setwb(net, parametry);

```

Obrázek 16: Část skriptu určená pro vytvoření a nastavení neuronové sítě

## 6. Vlastní práce – spotřeba tepla

V této kapitole jsou popsány postupy vypracování úkolů diplomové práce týkajících se spotřeby tepla.

### 6.1. Data pro výpočet spotřeby tepla

Dalším krokem práce bylo se pokusit o predikci spotřeby tepla objektu neboli množství tepla odebraného ze sítě dálkového vytápění. Pro tuto část práce byla použita data naměřená při výzkumu prováděném Ústavem mechaniky tekutin a termodynamiky Českého vysokého učení technického v Praze, viz [12]. Tento výzkum se zabýval optimalizací sítě dálkového vytápění budov. Měření byla prováděna v několika budovách nacházejících se v obci Strakonice na sídlišti Mír. Pro tuto práci byla použita data, naměřená v jednom z těchto objektů.

V tomto odstavci bylo vycházeno z informací obsažených ve zdroji [13]. Množství odebraného tepla  $Q$  se získá integrací tepelného výkonu  $P_q$  za časový interval od  $t_1$  do  $t_2$ :

$$Q = \int_{t_1}^{t_2} P_q dt \quad (5.5.1)$$

Vztah pro výpočet tepelného výkonu má tvar:

$$P_q = \dot{V} \cdot (T_1 \cdot c_{p1} \cdot \rho_1 - T_2 \cdot c_{p2} \cdot \rho_2) \quad (5.5.2)$$

kde  $\dot{V}$  je objemový průtok,  $T$  je teplota,  $\rho$  je hustota a  $c_p$  je měrná tepelná kapacita teplosměnné látky. Index 1 označuje médium vstupující do objektu a index 2 médium vystupující z objektu. Z tohoto vzorce vyplývá, že bylo třeba získat vstupní teplotu, výstupní teplotu a objemový průtok vody vytápějící vybranou budovu. Získaná data obsahovala záznamy obou teplot a objemu vody, která objektem od počátku měření do času záznamu protekla. Data obsahovala i časové údaje, kdy byly hodnoty zaznamenány, což se dělo vždy po třech minutách. Díky tomu mohly být pomocí následujícího vzorce vypočítány hodnoty objemového průtoku.

$$\dot{V} = \frac{(V_2 - V_1)}{t} \quad (5.5.3)$$

$V_1$  značí hodnotu objemu na počátku časového intervalu,  $V_2$  je hodnota objemu na konci intervalu a  $t$  je délka časového intervalu.

V naměřených datech se nacházely několikadenní úseky, ve kterých nebyly hodnoty zaznamenávány. Protože bylo pro učení neuronové sítě potřeba dostatečné množství dat, byl vybrán jeden z nejdelších intervalů bez těchto výpadků a to 29. 3. 2021 až 31. 5. 2021. Protože spotřeba tepla objektu velmi závisí na venkovní teplotě, byly pro predikci zvolena data ze stejné části roku a to od 3. 4. 2020 do 28. 4. 2020. Přestože v těchto obdobích nenastaly žádné velké výpadky měření, chybělo v nich několik hodnot. Vždy se však jednalo o maximálně dva po sobě jdoucí záznamy, proto je bylo možné poměrně přesně nahradit. Přestože byl objem zaznamenáván každé tři minuty, měřen byl pouze jednou za dvacet minut. Chybějící hodnota se tedy vždy rovnala té předchozí nebo následující. Z hodnot objemů na začátku a na konci hodinového intervalu byl podle vzorce (5.5.3) vypočítán příslušný objemový tok. Chyběly-li hodnoty teploty, byly dopočítány lineární interpolací. Pokud nebyly známé dvě po sobě jdoucí hodnoty, byla první z nich určena na základě dvou předcházejících hodnot podle vzorce (5.5.4). V případě, že byla známá hodnota, která následovala po té chybějící, byla vypočítána jako průměrná hodnota lineárních interpolací ze dvou předchozích a ze dvou následujících záznamů podle vzorce (5.5.5). Tyto výpočty byly provedeny v programu MATLAB viz Obrázek 15.

$$T(i) = T(i - 1) + (T(i - 1) - T(i - 2)) \quad (5.5.4)$$

$$T(i) = \frac{T(i - 1) + (T(i - 1) - T(i - 2)) + T(i + 1) + (T(i + 1) - T(i + 2))}{2} \quad (5.5.5)$$

Poté, co byly doplněny všechny chybějící hodnoty, byly vstupní i výstupní teploty přepočítány na aritmetické průměry po hodinových intervalech. Následně bylo třeba určit hustotu a měrnou tepelnou kapacitu. Obě tyto hodnoty jsou obecně závislé na teplotě, ale protože se médium pohybovalo v relativně malém rozmezí teplot (22-59 °C), bylo podle zdroje [14] možné uvažovat měrnou tepelnou kapacitu jako konstantu  $c_p = 4183 \text{ J/kg}\cdot\text{K}$ . Na hustotu vody má teplota větší vliv, proto byla vytvořil Tabulka 6, podle které byla každé teplotě přiřazena hodnota hustoty. Nyní už mohl být proveden výpočet tepelného výkonu podle vzorce:

$$P_q = \dot{V} \cdot c_p \cdot (T_1 \cdot \rho_1 - T_2 \cdot \rho_2) \quad (5.5.6)$$

Ten vznikl úpravou vzorce (5.5.2). Hodnoty tepelných výkonů tak utvořily časovou řadu, která byla dále použita k predikci.

```
for i = 1:n
    if isnan(T1(i))
        if isnan(T1(i+1)) + isnan(T1(i+2)) > 0
            T1(i) = T1(i-1)+(T1(i-1)-T1(i-2));
        else
            T1(i) = ((T1(i-1)+(T1(i-1)-T1(i-2)))+(T1(i+1)+(T1(i+1)-T1(i+2))))/2;
        end
    end

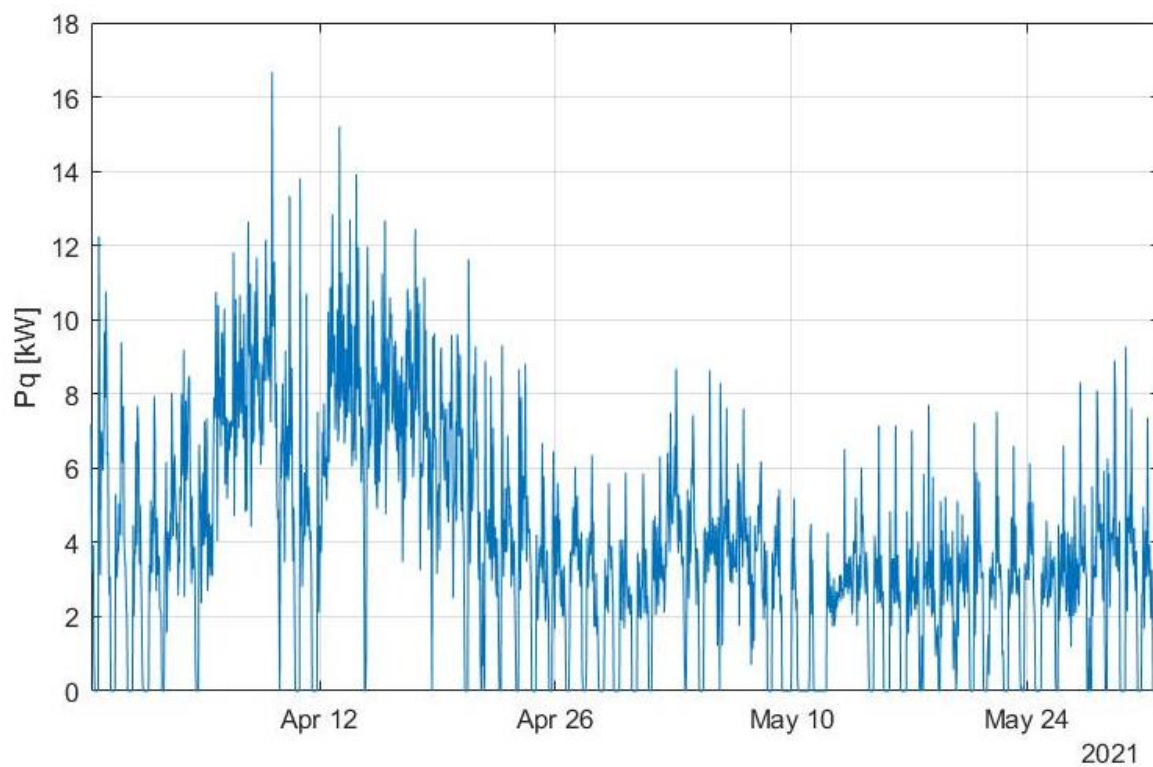
    if isnan(T2(i))
        if isnan(T2(i+1)) + isnan(T2(i+2)) > 0
            T2(i) = T2(i-1)+(T2(i-1)-T2(i-2));
        else
            T2(i) = ((T2(i-1)+(T2(i-1)-T2(i-2)))+(T2(i+1)+(T2(i+1)-T2(i+2))))/2;
        end
    end
end
```

Obrázek 17: Část skriptu určená pro výpočet chybějících hodnot teploty lineární interpolací, kde  $n$  je počet měření,  $T1$  je vstupní teplota a  $T2$  je výstupní teplota

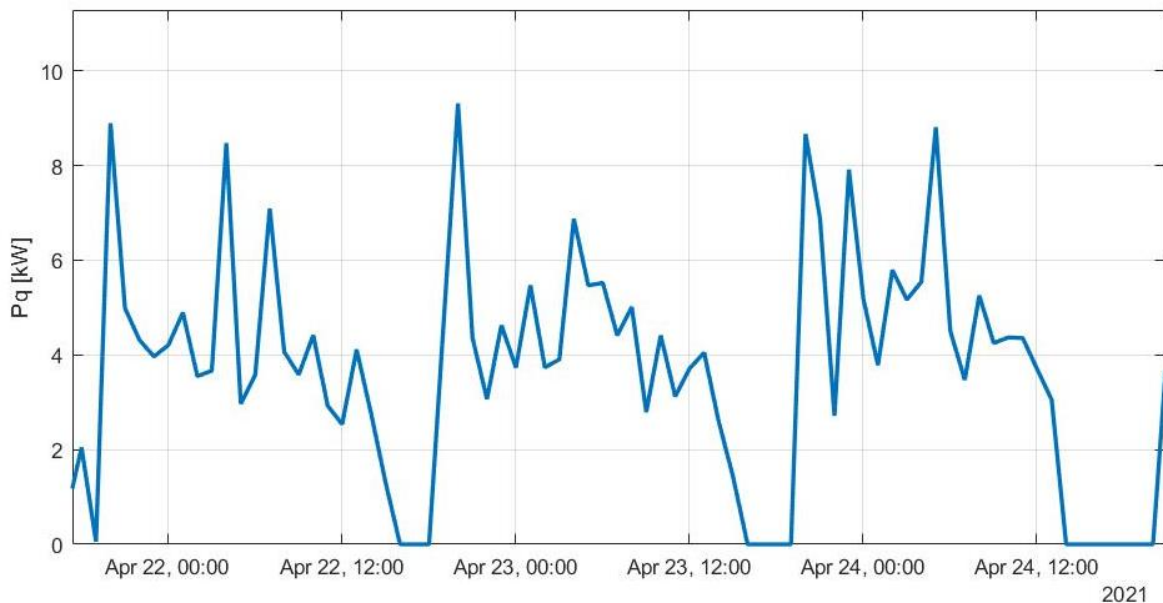
V grafu na je výsledná časová řada tepelné výkonu zobrazena. Zvláště pak na detailu na Obrázek 19 lze pozorovat že obsahuje denní cykly. Tepelný výkon je na začátku dne na vrcholu, kde zůstává celou noc a dopoledne. Na začátku odpoledne klesá až na nulovou hodnotu, ale brzdý zase začíná stoupat.

T [°C]	$\rho$ [kg/m <sup>3</sup> ]
22-23	998
24-27	997
28-30	996
31-33	995
34-36	994
37-39	993
40-41	992
42-44	991
45-46	990
47-48	989
49-51	988
52-53	987
54-55	986
56-57	985
58-59	984

Tabulka 6: Závislost hustoty vody  $\rho$  na teplotě  $T$



Obrázek 18: Graf zobrazující časovou řadu tepelného výkonu



Obrázek 19: Detail grafu na Obrázku 15

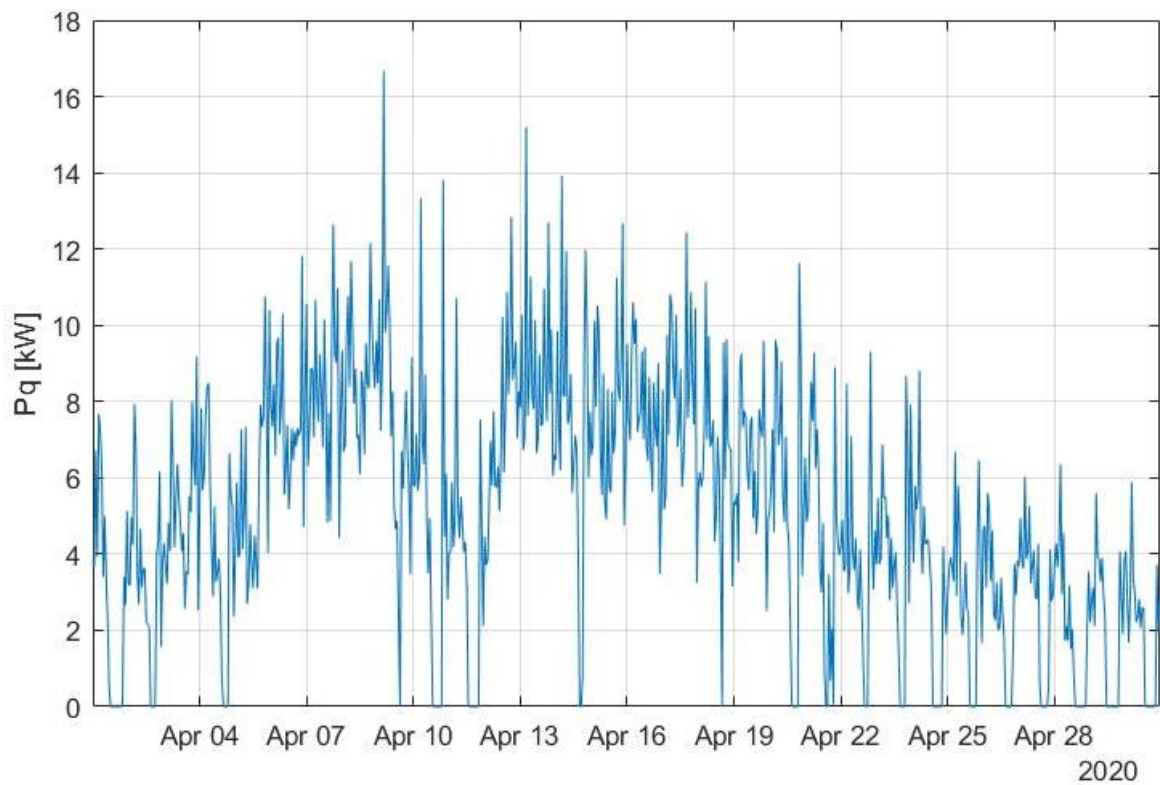
## 6.2. Predikce tepelného výkonu

Predikce tepelného výkonu byla provedena podobným způsobem jako predikce rychlosti větru. Nejprve byla v programu MATLAB provedena analýza časové řady určené pro učení neuronové sítě viz Obrázek 18. Pomocí příkazu `phaseSpaceReconstruction` byl sestaven fázový prostor, pro nějž byly vypočteny parametry dimenze vnoření  $m_d = 4$  a časové zpoždění  $\tau = 7$ . Pro predikci byla opět použita síť se dvěma skrytými vrstvami. Její učení probíhalo podle BR algoritmu a bylo provedeno dvacetkrát. Vybrány byly ty parametry sítě, které dosáhly při kontrole výsledků pomocí funkce `mse` nejnížší střední kvadratické odchylky:  $\sigma^2 = 1,99 \text{ kW}^2$ . Tyto parametry byly opět uloženy pomocí funkce `getwb`.

Jak už bylo zmíněno dříve, predikce byla prováděna na datech z dubna roku 2020 viz Obrázek 20 z toho důvodu, aby byl průběh odebíraného tepelného výkonu podobný datům, na kterých se neuronová síť učila. Z této časové řady byl rekonstruován fázový prostor s předem zadanými parametry  $m_d = 4$  a  $\tau = 7$ , aby odpovídal požadavkům neuronové sítě. Byl použit stejný mechanismus pro predikci větru viz Obrázek 15, který sloužil k tomu, aby byly hodnoty vypočítané neuronovou sítí dosazeny do fázového prostoru a použity jako vstupy pro další kroky predikce. Neuronová síť byla opět sestavena



stejným způsobem podle Obrázek 16. Byla vytvořena síť se dvěma skrytými vrstvami, které měly po patnácti neuronech, se čtyřmi vstupy a jedním výstupem podle vstupních a výstupních dat a váhy spojů a prahy neuronů byly dodány z procesu učení. Následně byla několikrát provedena predikce s počátkem v různých časech, aby bylo možné objektivně posoudit její efektivitu. Výsledky byly shrnuty v následující kapitole.



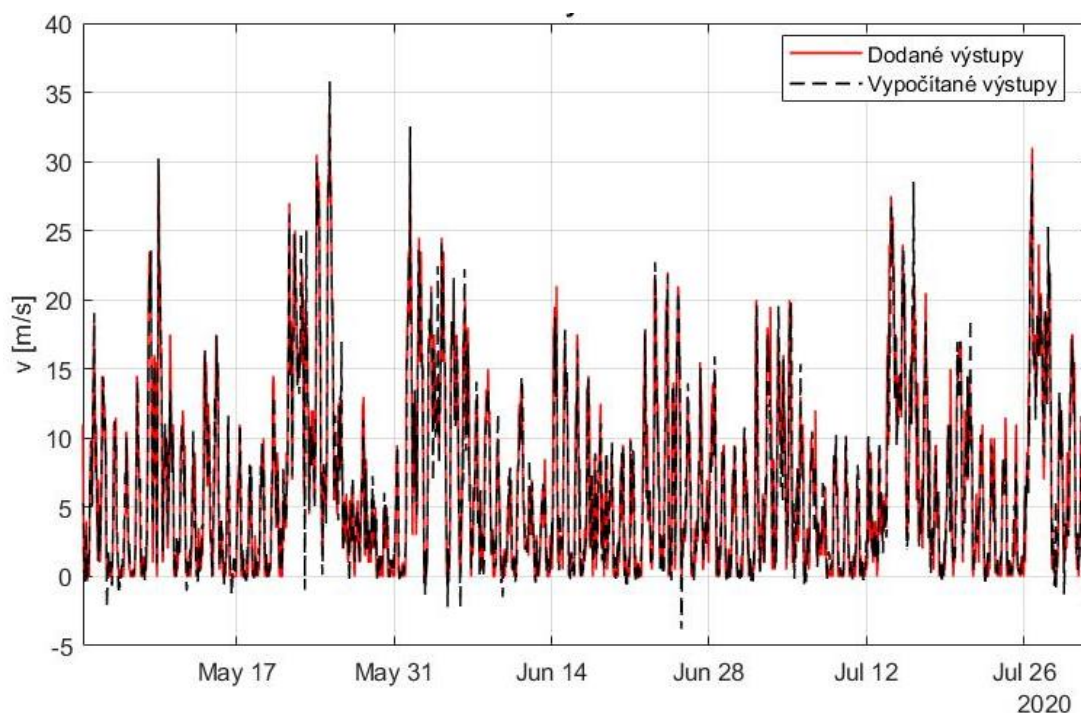
Obrázek 20: Graf zobrazující časovou řadu tepelného výkonu určeného pro predikci

## 7. Výsledky a diskuse

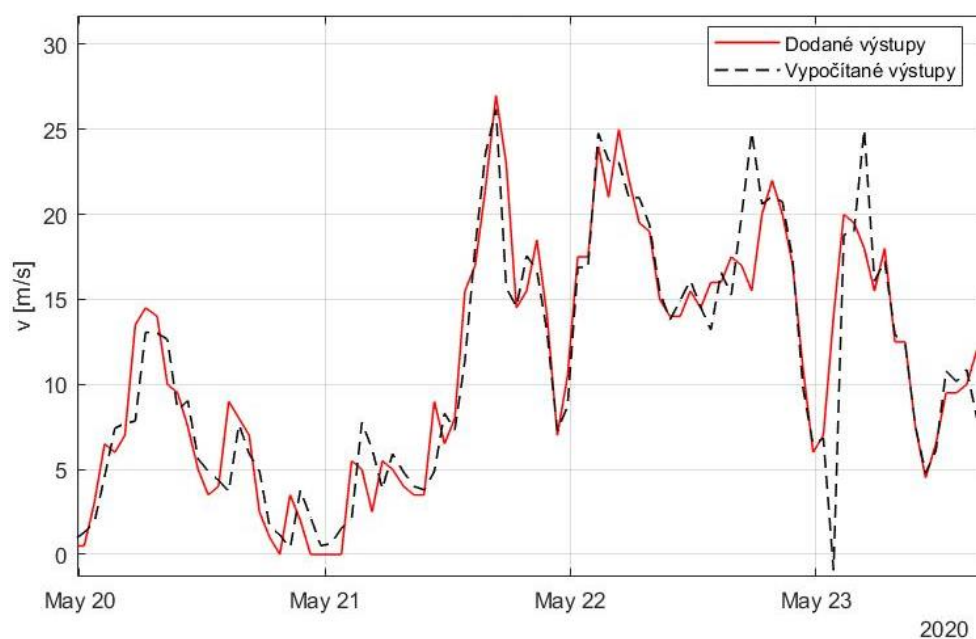
V této kapitole popisují výsledky predikce rychlosti větru a tepelného výkonu.

### 7.1. Predikce rychlosti větru

První predikce v této práci byla prováděna již v rámci procesu učení neuronových sítí. Byly vypočítávány střední kvadratické odchylky výsledků těchto predikcí od správných hodnot a na základě toho byly jednotlivé sítě hodnoceny. Jednalo se sice pouze o predikci následujícího kroku, přesto však byly její výsledky velmi zásadní, protože z nich bylo možné odhadnout, jak dopadne vícekroková predikce. Na Obrázek 21 byl zobrazen výsledek predikce celé časové řady. Z tohoto grafu se zdá, že byla predikce až na několik výjimek velice přesná, skutečnost lze však lépe posoudit z detailu na Obrázek 22. Zde bylo možné pozorovat, že predikce skutečně v některých bodech proběhla velmi úspěšně, jako tomu je například u hodnot naměřených kolem půlnoci 21. května. Poměrně často se ale stávalo, že se zdálo, že vypočítané výsledky spíše odpovídají předchozí hodnotě než té, kterou predikovaly. Jako tomu je například v první části dne 20. května. V některých případech se odhady zdánlivě bez příčiny odchýlily velmi výrazně, jako tomu je u hodnot naměřených hned po půlnoci 22. května. Všechny tři zmíněné úseky byly sepsány do Tabulka 7, kde jsou vyjádřeny i kvadratické odchylky v jednotlivých časech. Střední kvadratická odchylka hodnot v první části tabulky pak je  $\sigma^2 = 1,25 \text{ m}^2/\text{s}^2$ , ve druhé části  $\sigma^2 = 6,52 \text{ m}^2/\text{s}^2$  a ve třetí části  $\sigma^2 = 39,32 \text{ m}^2/\text{s}^2$ .



Obrázek 21: Predikce následující hodnoty rychlosti větru



Obrázek 22: Detail grafu na Obrázek 21

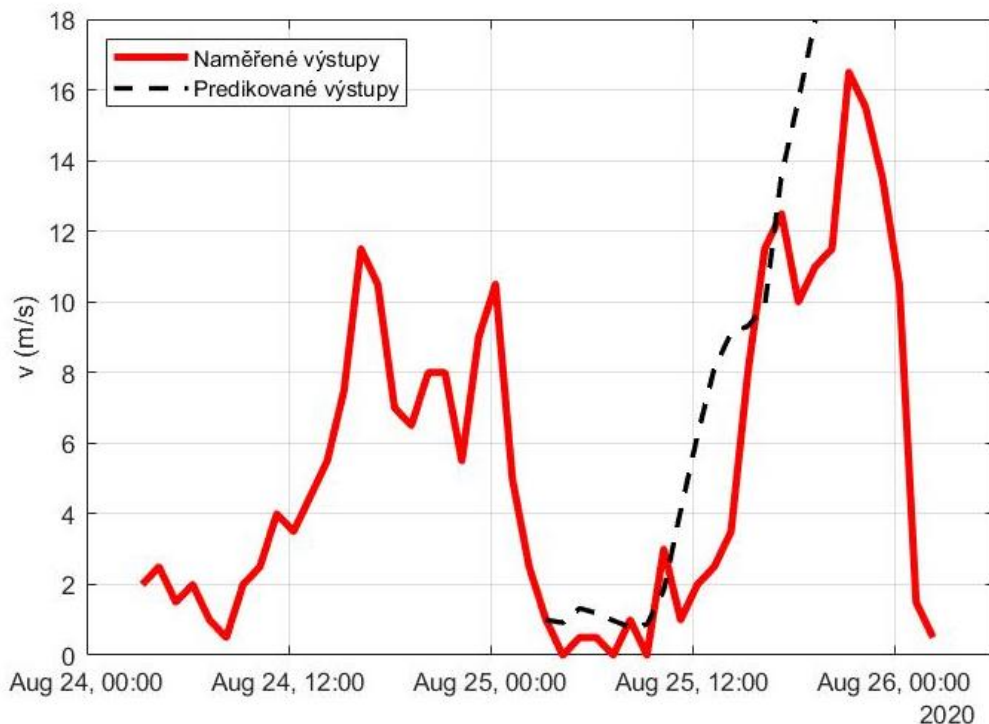
	21-May 20:43	21-May 22:43	21-May 22:43	21-May 23:43	22-May 00:43	22-May 01:43	22-May 02:43
$v_n$ [m/s]	18,5	14	7	10,5	17,5	17,5	24
$v_p$ [m/s]	16,70	13,04	7,26	8,72	16,88	16,90	24,79
$\sigma^2$ [m <sup>2</sup> /s <sup>2</sup> ]	3,24	0,93	0,07	3,17	0,39	0,36	0,62

	20-May 03:43	20-May 04:43	20-May 05:43	20-May 06:43	20-May 07:43	20-May 08:43	20-May 09:43
$v_n$ [m/s]	6	7	13,5	14,5	14	10	9,5
$v_p$ [m/s]	7,42	7,71	7,84	13,07	13,01	12,65	8,47
$\sigma^2$ [m <sup>2</sup> /s <sup>2</sup> ]	2,01	0,50	31,99	2,05	0,97	7,03	1,05

	23-May 00:43	23-May 01:43	23-May 02:43	23-May 03:43	23-May 04:43	23-May 05:43	23-May 06:43
$v_n$ [m/s]	7	14	20	19,5	18	15,5	18
$v_p$ [m/s]	6,84	-0,95	18,79	19,00	25,01	16,08	17,24
$\sigma^2$ [m <sup>2</sup> /s <sup>2</sup> ]	0,03	223,49	1,46	0,25	49,12	0,33	0,57

Tabulka 7: Výsledky jednokrokové predikce, kde  $v_n$  je naměřená rychlost,  $v_p$  je predikovaná rychlost a  $\sigma^2$  je střední kvadratická odchylka.

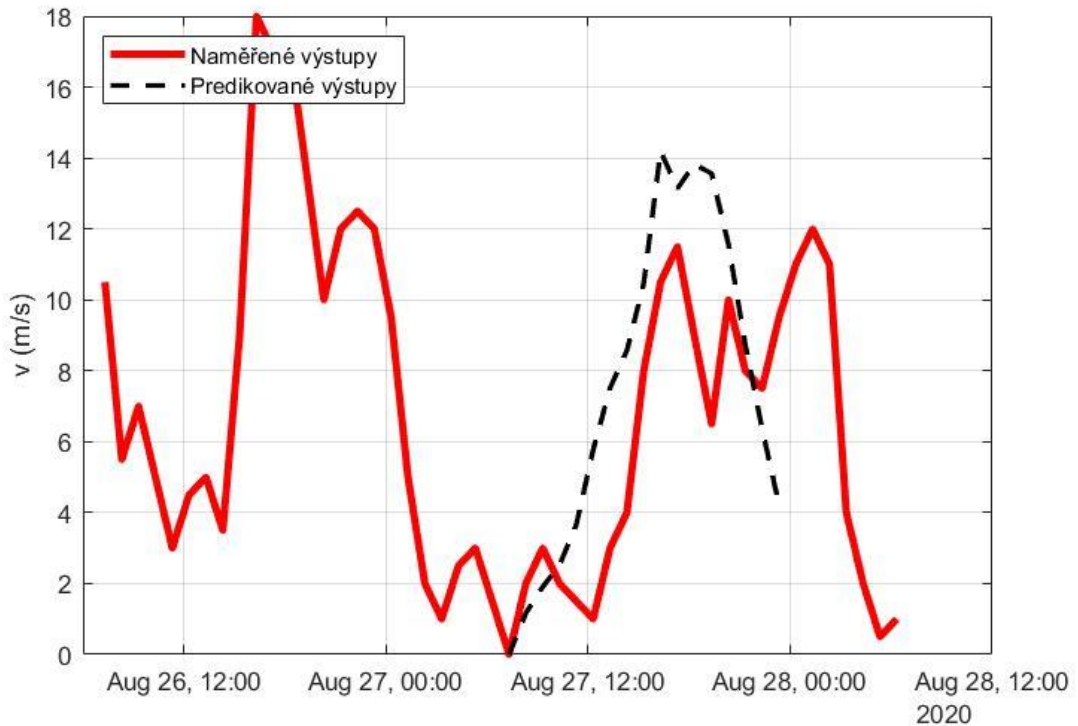
Predikce, která měla simulovat skutečnou předpověď rychlosti větru byla prováděna na hodnotách naměřených v srpnu 2020 viz Obrázek 14. Byly vybrány čtyři body nacházející se v různých částech denního cyklu hodnot časové řady, které byly použity jako výchozí body predikce. Byly zvoleny tak, aby se nacházely v různých částech denního cyklu rychlosti větru a neuronová síť tak byla aplikována na různé podmínky. První predikce, která byla provedena, počínala v části denního cyklu, kdy vítr právě přestal a následovala doba, kdy dosahoval pouze mírných rychlostí, následovaných poměrně prudkým nárůstem. V grafu na Obrázek 23 je možné pozorovat, že přesto že se směrnice grafu právě náhle změnila, neuronová síť to dokázala předpovědět. Následný nárůst sice předpovídala o několik hodin dříve, ale alespoň se jí přibližně podařilo predikovat budoucí trend. Nepodařilo se však odhadnout maximum rychlosti a byl stále předpovídán další růst. Pro prvních deset predikovaných hodnot byla vypočítána střední kvadratická odchylka  $\sigma^2 = 6,38 \text{ m}^2/\text{s}^2$ .



Obrázek 23: První víceokrová predikce rychlosti větru

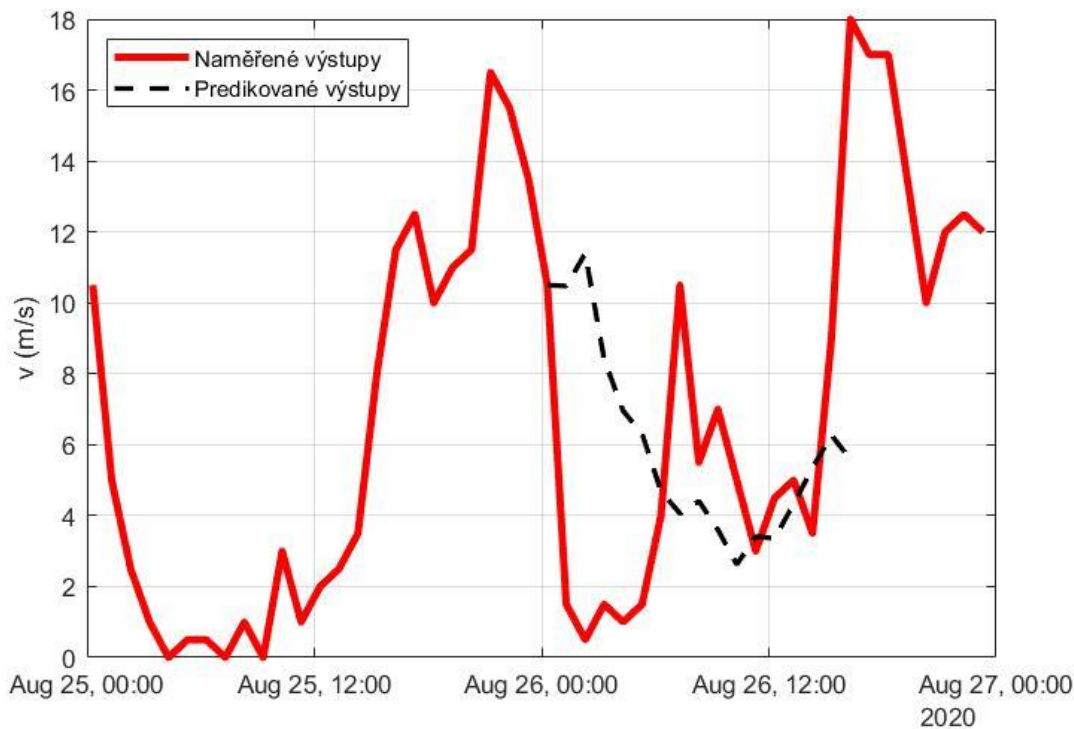
Výsledek druhé predikce je možné pozorovat na Obrázek 24. V tomto případě byl zvolen počátek v čase, kdy je hodnota rychlosti větru nulová, nachází se totiž v minimu denního cyklu. Podle naměřených dat hodnota následuje mírný nárůst, po něm mírný pokles a následně výrazný nárůst vedoucí k vysokým hodnotám rychlosti. Neuronová síť

však předpovídá tento prudký nárůst hned vzápětí. Podaří se jí však sledovat trend grafu a maximum rychlosti určí přibližně ve stejném čase, jako i ve skutečnosti nastane. Pro prvních deset predikovaných hodnot byla určena střední kvadratická odchylka  $\sigma^2 = 9,33 \text{ m}^2/\text{s}^2$ .



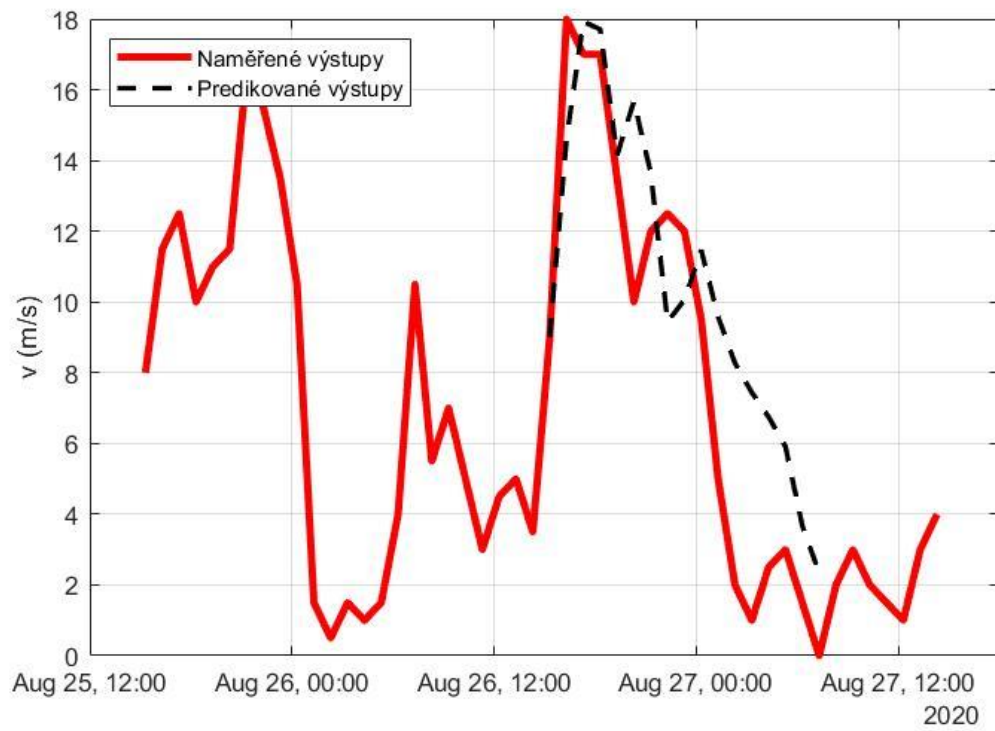
Obrázek 24: Druhá vícezkroková predikce rychlosti větru

Třetí predikce byla zaznamenána na Obrázek 25. Počátek této predikce leží v momentě, kdy dochází k prudkému poklesu rychlosti větru až na hodnoty blízké nule. Ty jsou však brzy následovány výrazným nárůstem. V tomto případě byla neuronová síť v predikci výrazně méně úspěšná. Po dvou krocích sice předpovídala pokles, ale ten byl pouze mírný a trend sledovala pouze přibližně. Vůbec však nedokázala předpovědět následný nárůst. Je nutno uznat, že tento vzrůst rychlosti se vymyká typickému dennímu cyklu rychlosti větru, což bude pravděpodobně příčinnou, proč ho neuronová síť nedokázala predikovat. Střední kvadratická odchylka pro prvních deset predikovaných hodnot činila  $\sigma^2 = 36,58 \text{ m}^2/\text{s}^2$ , což je příliš velká hodnota, aby bylo možné tuto predikci považovat za úspěšnou.



Obrázek 25: Třetí víceokrová predikce rychlosti větru

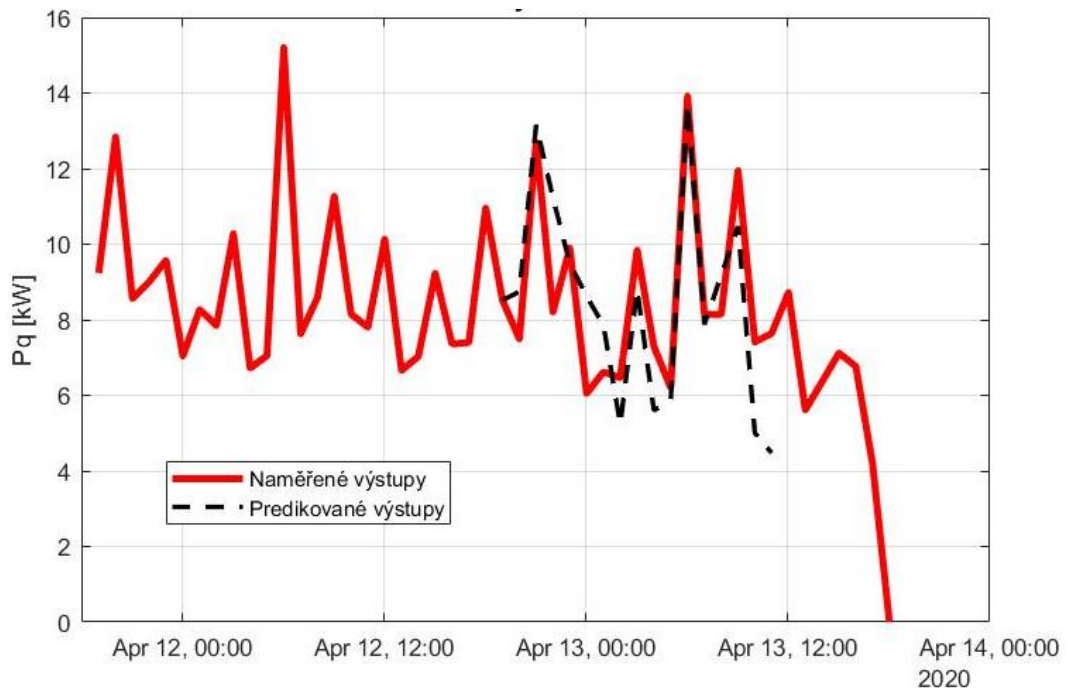
Čtvrtá predikce, která je zobrazena na Obrázek 26, začíná v momentu, kdy rychlost větru strmě narůstá, ale už v prvním kroku dosáhne vrcholu a začíná klesat. Neuronová síť v tomto případě sice předpokládala dosažení maxima o krok později, ale jeho hodnotu určila velmi přesně. I výsledky vypočítané pro dva další kroky se velmi podobají správným výsledkům, ale dále už se hodnoty počínají více rozcházet. Předpověď alespoň stále sleduje klesající trend naměřených hodnot. Pro prvních deset predikovaných hodnot byla vypočítána střední kvadratická odchylka  $\sigma^2 = 6,38 \text{ m}^2/\text{s}^2$ .



Obrázek 26: Čtvrtá vícezkrová predikce rychlosti větru

## 7.2. Predikce tepelného výkonu

Bylo rozhodnuto, že predikce tepelného výkonu bude prováděna na datech naměřených v dubnu 2020 čili o rok dříve ale ve stejné části roku, jako byly naměřeny data pro učení neuronové sítě. Na spotřebu tepla má totiž zásadní vliv venkovní teplota a touto volbou bylo zamýšleno získat co nejvíce podobné podmínky. Stejně jako u predikce rychlosti větru byly pro předpověď tepelného výkonu vybrány čtyři body ležící v různých částech denního cyklu časové řady. První predikce viz Obrázek 27 provedena pro hodnoty naměřené v čase, kdy data nevykazovala typický denní cyklus a tepelný výkon v průběhu dne stále kolísal ve stejném rozmezí hodnot. Přesto se neuronové síti podařilo sestavit relativně přesnou predikci, která stále sledovala trend grafu naměřených hodnot a dokázala určit maxima a minima ve správných krocích. Střední kvadratická odchylka pro prvních deset kroků činila  $\sigma^2 = 2,48 \text{ kW}^2$ .

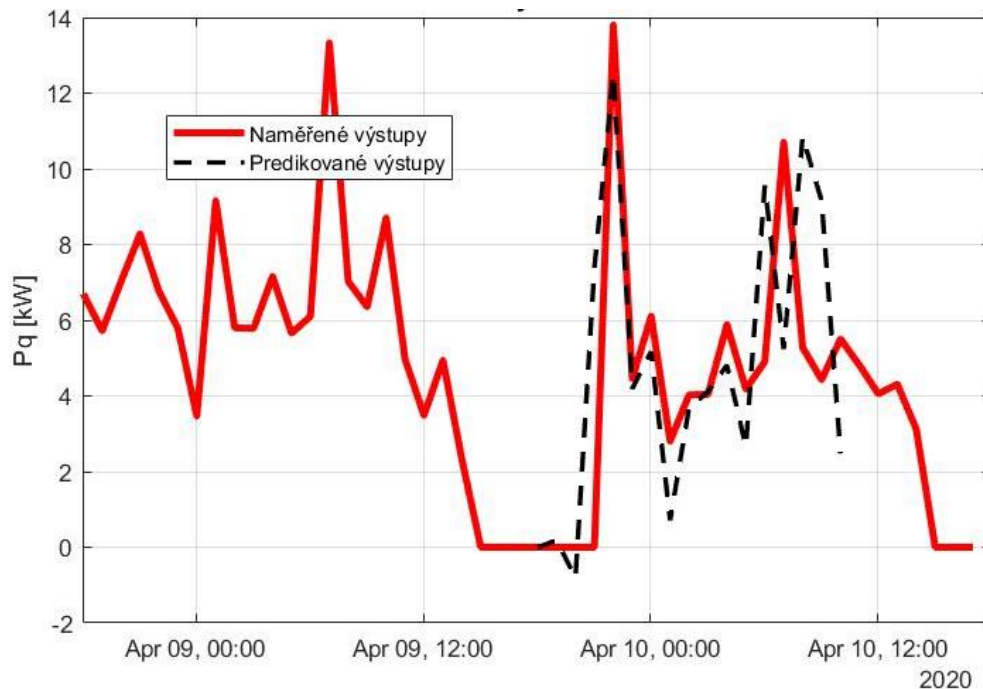


Obrázek 27: První vícekroková predikce tepelného výkonu

Počátek druhé predikce viz Obrázek 28 byl zvolen v kroku, kdy se hodnota teplotního cyklu nachází na minimu denního cyklu a pro několik dalších kroků zůstává rovna nule. Poté následuje velmi prudký nárůst následovaný o něco menším poklesem, po

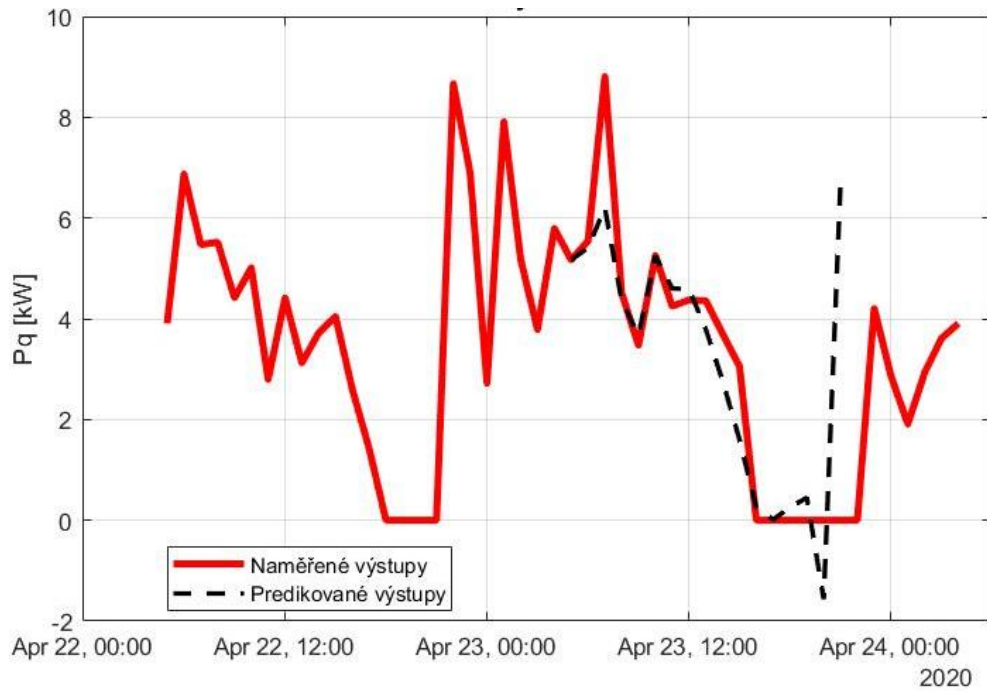


kterém se tepelný výkon ustálí v rámci běžných hodnot. Neuronová síť v tomto případě provedla predikci velmi přesně. Dokázala odhadnout obě velké změny hodnot a dařilo se jí předpovídat i následující průběh. Chybou však bylo, že nárůst předpověděla již o krok dřív, což zapříčinilo mnohem vyšší průměrnou střední kvadratickou odchylku, než by jinak v tomto případě byla. Tentokrát byla vypočítána jako  $\sigma^2 = 6,34 \text{ kW}^2$ .



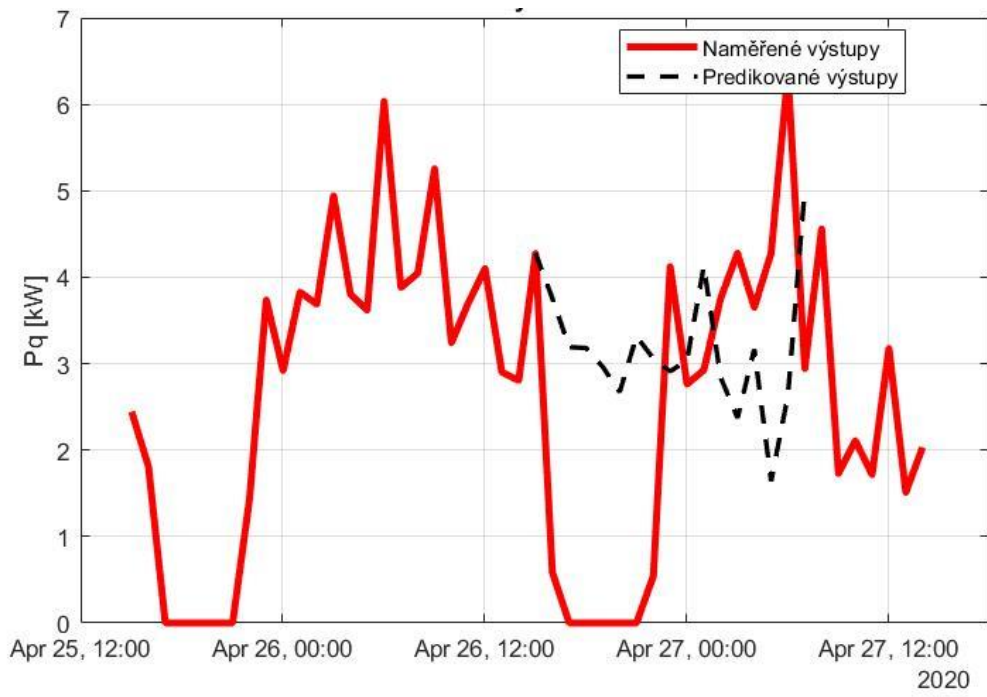
Obrázek 28: Druhá vícekroková predikce tepelného výkonu

Výsledek třetí predikce je zobrazen v grafu na Obrázek 29. V tomto případě byl jako počátek predikce vybrán krok nacházející se uprostřed horní části denního cyklu. Po něm následuje nárůst na denní maximum a poté už pomalý pokles hodnot. Neuronové síti se povedlo provést predikci velmi přesně, pouze nedokázala správně určit hodnotu denního maxima. Všechny ostatní z prvních devíti hodnot předpověděla se střední kvadratickou odchylkou menší než  $0,5 \text{ kW}^2$ . Výsledná střední odchylka pro prvních deset hodnot pak činila  $\sigma^2 = 1,04 \text{ kW}^2$ .



Obrázek 29: Třetí vícezkroková predikce tepelného výkonu

Čtvrté provedení predikce tepelného výkonu je vyobrazeno na . V tomto případě byl počátek predikce umístěn do moment, kdy je hodnota výkonu stále poměrně vysoká, ale vzápětí v souladu s denním cyklem prudce klesá až k nulovým hodnotám. Předpověď, kterou sestavila neuronová síť tomuto průběhu však vůbec neodpovídá. Není v ní patrný žádný prudký pokles. Hodnoty tepelného výkonu se pohybují ve stále stejných rozmezích jako v části dne, kdy je spotřeba tepla nejvyšší. Výsledná střední kvadratická odchylka tak byla určena jako  $\sigma^2 = 6,66 \text{ kW}^2$ .



Obrázek 30: Čtvrtá víceřádková predikce teplotního výkonu

## 8. Závěr

Prvním cílem této práce bylo analyzovat chaotickou časovou řadu z pohledu potenciálu pro výrobu elektrické energie za použití větrné elektrárny. Data byla získána z portálu thingspeak, následně byla upravena a přepočítána na hodnoty ve výšce, kde by se nacházel střed rotoru turbíny. V dalším kroku byla vyobrazena podle Weibullova rozdělení pravděpodobnosti. Z tohoto vyobrazení bylo zjištěno, že místo, ve kterém bylo měření prováděno není pro vybudování větrné elektrárny optimální. Často zde totiž vítr fouká malou rychlostí nebo nefouká vůbec, takže by turbína nemohla být spuštěná. Pro výpočet výkonu, který by bylo možné pomocí energie větru vytvořit, byla vybrána turbína V126 – 3,45 MW od firmy Vestas. S pomocí výkonové křivky této turbíny byl z Weibullova grafu vypočten průměrný výkon turbíny, který činil 1521,16 kW. Což odpovídá 44 % maximálního dosažitelného výkonu této turbíny.

Dalším cílem této práce bylo analyzovat chaotickou časovou řadu rychlosti větru pro rekonstrukci fázového prostoru. To bylo provedeno v programu MATLAB pomocí funkce phaseSpaceReconstruction. Výsledkem této analýzy bylo určení parametrů fázového prostoru, kdy dimenze vnoření vyšla jako  $m_d = 5$  a časové zpoždění jako  $\tau = 8$ . Následně byla vytvořena dopředná neuronová síť. Nejlepších výsledků dosahovala síť se dvěma skrytými vrstvami obsahujícími každá po patnácti neuronech, jejíž učení bylo provedeno podle algoritmu Bayesianovi regularizace. Tato síť dosahovala predikce na jeden krok dopředu se střední kvadratickou odchylkou  $\sigma^2 = 4,05 \text{ m}^2/\text{s}^2$ . Byl vybrán úsek predikované časové řady, ve kterém se nacházely oblasti se třemi typickými výsledky predikce. V první z nich dosahovala neuronová síť přesných výsledků a odchylka byla vypočtena jako  $\sigma^2 = 1,25 \text{ m}^2/\text{s}^2$ . Ve druhé oblasti se zdálo, jako kdyby neuronová síť pouze opisovala předchozí hodnoty. Zde vyšla  $\sigma^2 = 6,52 \text{ m}^2/\text{s}^2$ . Ve třetí oblasti se predikce několikrát velmi odchýlila od správných výsledků, a proto byla určena  $\sigma^2 = 39,32 \text{ m}^2/\text{s}^2$ .

Cílem ale bylo provést zdárně predikci budoucího vývoje rychlosti větru na alespoň několik hodin dopředu. Byly provedeny predikce pro čtyři různé části časové řady. Byly voleny takovým způsobem, aby byl počáteční krok vždy v jiné fázi denního cyklu a aby byl odlišný vývoj budoucího trendu hodnot rychlosti větru. Ve třech ze čtyř případů by se predikce dala označit za zdařilou. Byla vypočítána střední kvadratická odchylka pro prvních deset predikovaných hodnot. Neuronová síť byla schopna správně předpovědět trend budoucích hodnot rychlosti i na více než deset kroků dopředu. V několika případech byla také úspěšně určena hodnota lokálního extrému.

Druhá část této práce byla zaměřena na analýzu a predikci spotřeby tepla objektu. Pro usnadnění bylo pracováno s hodnotami tepelného výkonu, z nichž lze následně

integrací vypočítat spotřebu tepla. Potřebná data byla získána z měření výzkumu prováděného ústavem mechaniky tekutin a termodynamiky strojní fakulty ČVUT. Z naměřených hodnot tepla a objemu byla vypočtena časová řada tepelného výzkumu. Ta byla následně analyzována v programu MATLAB pomocí funkce `phaseSpaceReconstruction`. Touto analýzou byly získány parametry fázového prostoru dimenze vnoření, resp. časové zpoždění, které byly určeny jako  $m_d = 4$ , resp.  $\tau = 7$ . Následně byla provedena predikce pomocí neuronové sítě, která opět obsahovala dvě skryté vrstvy po patnácti neuronech a byla učena podle BR algoritmu. Predikce tepelného výkonu byla úspěšně provedena ve třech ze čtyř pokusů. Neuronové sítě se až na jeden případ dařilo velmi přesně predikovat budoucí vývoj hodnot i přes to, že se trend prudce měnil či se nejednalo a vývoj typický pro denní cyklus výkonu.

Lepších výsledků by jistě bylo možné dosáhnout a možností, jak toho docílit, se nabízí hned několik. Jednou z nich je získání dat pomocí přesnějších měření, dále použití většího rozsahu dat pro učení neuronové sítě a v neposlední řadě použití lepší neuronové sítě.

## 9. Literatura

- [1] DLOUHÝ, Martin. *Datová analýza a simulátor rychlosti větru z veřejně dostupných dat*. Plzeň, 2014. Diplomová práce. Západočeská univerzita v Plzni, Fakulta aplikovaných věd, Katedra kybernetiky.
- [2] Help Center. *MathWorks* [online]. Natick (Massachusetts): MathWorks, ©1994-2022 [cit. 2022-01-04]. Dostupné z: <https://www.mathworks.com/help/index.html>
- [3] SNELL, Richard S. *Clinical Neuroanatomy*. 7th edition. Philadelphia: Lippincott Williams & Wilkins, 2010. ISBN 978-0-7817-9427-5.
- [4] SKOŘEPA, Jan. *Základní postupy analýzy chaotických signálů založených na teorii dynamických systémů*. Praha, 2010. Bakalářská práce. Univerzita Karlova v Praze, Matematicko-fyzikální fakulta, Katedra meteorologie a ochrany prostředí.
- [5] GAO, Zhongke a Ningde JIN. Complex network from time series based on phase space reconstruction. *Chaos: An Interdisciplinary Journal of Nonlinear Science* [online]. 2009, **19**(3) [cit. 2021-12-04]. ISSN 1054-1500. Dostupné z: doi:10.1063/1.3227736
- [6] DONGARE, A.D., R.R. KHARDE a A.D. KACHARE. Introduction to Artificial Neural Network. *International Journal of Engineering and Innovative Technology* [online]. July 2012, **2**(1), 189-194 [cit. 2021-12-04]. ISSN: 2277-3754. Dostupné z: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.1082.1323&rep=rep1&type=pdf>
- [7] ABIODUN, Oludare Isaac, Aman JANTAN, Abiodun Esther OMOLARA, Kemi Victoria DADA, Nachaat AbdElatif MOHAMED a Humaira ARSHAD. State-of-the-art in artificial neural network applications: A survey. *Heliyon* [online]. 2018, **4**(11) [cit. 2022-01-04]. ISSN 24058440. Dostupné z: doi:10.1016/j.heliyon.2018.e00938
- [8] *Danish wind industry association* [online]. Copenhagen, ©1997-2003 [cit. 2021-12-04]. Dostupné z: <http://xn--drmstrre-64ad.dk/wp-content/wind/miller/windpower%20web/en/tour/wres/wndsprac.htm>

- [9] WAGNER, Rozenn, Ioannis ANTONIOU, Søren M. PEDERSEN, Michael S. COURTNEY a Hans E. JØRGENSEN. The influence of the wind speed profile on wind turbine performance measurements. *Wind Energy* [online]. 2009, **12**(4), 348-362 [cit. 2022-01-04]. ISSN 10954244. Dostupné z: doi:10.1002/we.297
- [10] MOHANDÉS, Mohamed A., Shafiqur REHMAN a Talal O. HALAWANI. A neural networks approach for wind speed prediction. *Renewable Energy* [online]. 1998, **13**(3), 345-354 [cit. 2022-01-04]. ISSN 09601481. Dostupné z: doi:10.1016/S0960-1481(98)00001-9
- [11] V126-3.45 MW. *Vestas* [online]. Aarhus (Dánsko), ©2021 [cit. 2021-12-05]. Dostupné z: <https://www.vestas.com/en/products/4-mw-platform/V126-3-45-MW>
- [12] SLÁMA, P. a J. NOŽIČKA. Implementing Cloud Solutions to Optimize the Operation of District Heating Networks with Preparation for MindSphere®. *Acta Mechanica Slovaca* [online]. 2021, **25**(1), 42-50 [cit. 2022-01-05]. ISSN 13352393. Dostupné z: doi:10.21496/ams.2021.007
- [13] KADLEC, Karel. Měření množství tepla. *Automa: časopis pro automatizační techniku*. 2016, **22**(4), 8-10. ISSN 1210-9592.
- [14] JAHODA, M., O. HOLEČEK a L. SCHREIBEROVÁ. E-tabulky: Fyzikální vlastnosti látek. *Ústav chemického inženýrství VŠCHT Praha* [online]. [cit. 2021-12-05]. Dostupné z: <https://uchi.vscht.cz/studium/tabulky>

## 10. Seznam obrázků

Obrázek 1: Výkonová křivka turbíny, na vodorovné ose je zobrazována rychlost větru a na svislé ose výkon turbíny [8].....	15
Obrázek 2: Weibullův graf, na svislé ose je zobrazena rychlost větru a na svislé ose pravděpodobnost, že vítr bude danou rychlostí foukat [8].....	16
Obrázek 3: Příklad použití funkce pro nahrávání dat z portálu Thinspeak.....	29
Obrázek 4: Data naměřená za jeden kalendářní rok získaná z portálu z Thingspeak .....	30
Obrázek 5: Weibullův graf pro rychlost větru ve výšce 10 m rozdělenou po 0,5 m/s .....	31
Obrázek 6: Výkonová křivka turbíny V126 – 3,45 MW Vestas .....	33
Obrázek 7: Weibullové grafy, A - ve výšce 87 m, B - ve výšce 117 m, C - ve výšce 137 m, D - ve výšce 147 m.....	34
Obrázek 8: Data rychlosti větru použítá pro učení neuronových sítí .....	35
Obrázek 9: Detail grafu na Obrázku 8 (datum na vodorovné ose značí začátek dne).....	36
Obrázek 10: Příkaz pro rekonstrukci fázového prostoru, kde data je časová řada, XR je fázový prostor, lag je časové zpoždění fázového prostoru ( $\tau$ ) a dim je počet dimenzí fázového prostoru ( $m_d$ ).....	36
Obrázek 11: Příkaz pro vytvoření dopředné neuronové sítě, kde hiddenSizes je vektor udávající počet skrytých vrstev a množství jejich neuronů, trainFnc udává, jako algoritmus bude použit při učení sítě a net je vytvořená neuronová síť.....	37
Obrázek 12: Část skriptu určená pro trénink a hodnocení výkonnosti neuronových sítí, kde net a NN jsou neuronové sítě, vstupy jsou vstupy neuronových sítí, cile jsou dodané výstupy, vystup jsou výstupy určené sítí a perfs jsou hodnocení výkonnosti sítí .....	38
Obrázek 13: Příkaz pro získání parametrů neuronové sítě, kde net je neuronová síť a parametry jsou její váhy a prahy .....	39
Obrázek 14: Data rychlosti větru použítá pro predikci pomocí neuronové sítě .....	40
Obrázek 15: Část skriptu určená pro predikci na N kroků dopředu, kde lag je časové zpoždění fázového prostoru $\tau$ a dim je počet jeho dimenzí $m_d$ , cile jsou dodaná výstupní data, vystup jsou vypočítaná výstupní data a vstupy jsou vstupní data .....	41



Obrázek 16: Část skriptu určená pro vytvoření a nastavení neuronové sítě .....	41
Obrázek 17: Část skriptu určená pro výpočet chybějících hodnot teploty lineární interpolací, kde $n$ je počet měření, $T_1$ je vstupní teplota a $T_2$ je výstupní teplota .....	44
Obrázek 18: Graf zobrazující časovou řadu tepelného výkonu .....	45
Obrázek 19: Detail grafu na Obrázku 15 .....	46
Obrázek 20: Graf zobrazující časovou řadu tepelného výkonu určeného pro predikci .....	47
Obrázek 21: Predikce následující hodnoty rychlosti větru .....	48
Obrázek 22: Detail grafu na Obrázku 20 .....	49
Obrázek 23: První vícekroková predikce rychlosti větru .....	50
Obrázek 24: Druhá vícekroková predikce rychlosti větru .....	51
Obrázek 25: Třetí vícekroková predikce rychlosti větru .....	52
Obrázek 26: Čtvrtá vícekroková predikce rychlosti větru .....	53
Obrázek 27: První vícekroková predikce tepelného výkonu .....	54
Obrázek 28: Druhá vícekroková predikce tepelného výkonu .....	55
Obrázek 29: Třetí vícekroková predikce tepelného výkonu .....	56

## 11. Seznam tabulek

Tabulka 1: Tabulka zobrazující třídy drsnosti a hodnoty délek drsnosti k příslušným typům krajiny převzato a upraveno z [8] .....	14
Tabulka 2: Dostupné algoritmy pro učení neuronových sítí v programu MATLAB, převzato a upraveno z [2] .....	28
Tabulka 3: Data pro výkonovou křivku turbíny V126 - 3,45 MW Vestas.....	32
Tabulka 4: Průměrný výkon větrné elektrárny .....	34
Tabulka 5: Srovnání přesnosti predikce různých sítí a algoritmů učení pomocí směrodatné kvadratické odchylky .....	39
Tabulka 6: Závislost hustoty vody $\rho$ na teplotě $T$ .....	45
Tabulka 7: Výsledky jednokrokové predikce, kde $v_n$ je naměřená rychlost, $v_p$ je predikovaná rychlost a $\sigma^2$ je střední kvadratická odchylka. ....	49