

České vysoké učení technické v Praze
Fakulta elektrotechnická

Disertační práce

říjen, 2013

Ing. Tomáš Pohl

České vysoké učení technické v Praze
Fakulta elektrotechnická
Katedra elektrotechnologie

Prediktivní technická diagnostika technologických procesů

Disertační práce

Ing. Tomáš Pohl

Praha, říjen, 2013

Doktorský studijní program: Elektrotechnika a informatika
Studijní obor: 2602V009 Elektrotechnologie a materiály

Školitel: Doc. Ing. Stanislav Seberský, CSc.

Školitel specialista: Ing. Ladislav Šmejkal, CSc.

Prohlášení

Prohlašuji, že jsem svou disertační práci zpracoval samostatně a v předložené práci důsledně citoval použitou literaturu.

V Praze dne

.....

podpis

Klíčová slova

Predikce, prediktivní diagnostika, technologický proces, konečný automat, model technologického procesu.

Keywords

Prediction, predictive diagnostics, technological process, a finite state machine, model of technological process.

Abstrakt

Disertační práce se věnuje prediktivní technické diagnostice technologických procesů. Popisuje dosavadní teorie a možnosti v metodách predikce, také se zmiňuje o technické diagnostice a tvorbě modelů. Dále se práce zabývá teorií konečných automatů, kterou nově rozšiřuje a doplňuje o data potřebná pro vytvoření konečného automatu pro prediktivní diagnostiku.

Popisované poznatky jsou v práci shrnuty do tvorby modelů technologického procesu založených na teorii konečných automatů a využitelných pro prediktivní diagnostiku technologického procesu. Popsaná teorie je v práci uvedena na vysvětlujících příkladech.

Abstract

This thesis deals with predictive technical diagnostics of technological processes. Thesis describes the current theories and possibilities in methods of prediction, mentions the technical diagnostics and model creation. The work deals with the finite state machine theory, the newly expanded and supplemented this theory by necessary data for the finite state machine for predictive diagnostics.

Data mentioned above are summarized in the creation of the technological process models based on the theory of finite state machine and are usable for predictive diagnostics and process control. Theory is described on two examples in this work.

OBSAH:

1. Úvod	8
1.1. Formulace problému.....	8
1.2. Současný stav problematiky a přínos disertační práce	8
1.2.1. Úvod	8
1.2.1. Současný stav	9
1.2.1.1. Diagnostika.....	10
1.2.1.1. Konečné automaty	14
1.2.2. Přínos disertační práce.....	16
1.3. Cíle disertační práce	18
2. Prediktivní diagnostika.....	20
2.1. Základní pojmy.....	20
2.2. Predikce	21
2.3. Prediktivní analýza	22
2.4. Prediktivní diagnostika pro údržbu	22
3. Prediktivní diagnostika technologických procesů	23
3.1. Metody predikce spojitých veličin	23
3.2. Modely.....	26
3.2.1. Prediktivní řízení založené na Modelu (MPC).....	28
3.2.2. Postupy při modelování a identifikaci	29
3.3. Diagnostické hybridní systémy	31
3.3.1. Konečný automat a technologický proces	31
3.3.2. Teorie konečných automatů	31
3.3.3. Konečný automat pro diagnostiku.....	37
3.3.4. Konečný automat pro prediktivní diagnostiku	40
3.3.4.1. Příklad konečného automatu pro prediktivní diagnostiku.....	43

3.4. Konečný automat pro prediktivní diagnostiku jako model technologického procesu .	49
3.4.1. Návrh modelu technologického procesu	49
3.4.1.1. Příklad technologického procesu 1 – Výroba produktu „A“	50
3.4.1.1.1. Tvorba modelu.....	51
3.4.1.2. Příklad technologického procesu 2 – Výroba maziva mícháním	59
3.4.1.2.1. Tvorba modelu.....	61
3.4.2. Funkce modelu	69
3.4.2.1. Parametry a výstupy modelu podle Příkladu 1	70
3.4.2.2. Parametry a výstupy modelu podle Příkladu 2.....	78
4. Závěr	83
5. Literatura	84

Seznam použitých symbolů

E	množina chybových stavů konečného automatu
J	kritérium optimalizace
O	množina výstupů konečného automatu - výstupní abeceda
P	prediktor
Q	množina stavů konečného automatu
q_E	poruchový stav konečného automatu, $q_E \in Q$
q_i	i -tý stav konečného automatu, $q_i \in Q$
q_0	počáteční stav konečného automatu, $q_0 \in Q$
s	počet stavů technologického procesu
$s(n)$	sledovaná veličina v predikci
T_{max}	množina maximálních časů
T_{min}	množina minimálních časů
u	vstupy do modelu soustav
x	stavy modelu soustavy
$x(t)$	vstupní proměnná
(x_i, y_i)	souřadnice i -tého bodu
y	výstupy modelu soustavy
$y(t)$	výstupní proměnná
y_p	predikovaná hodnota výstupní veličiny
$ w $	délka slova
α	časová řada událostí pro predikci
α_{T_i}	časová řada překročení času T_{max} i -té události v prediktivním konečném automatu
α_i	časová řada obecné i -té události v prediktivním konečném automatu
α_{TC}	počet uzlů diagnostikovaných časově
δ	přechodová funkce konečného automatu, $\delta: Q \times \Sigma \rightarrow Q$
δ_i	přechodová funkce přechodu ze stavu $i-1$ do stavu i
δ_{iE}	přechodová funkce přechodu ze stavu $i-1$ do poruchového stavu E
ε	prázdné slovo, prázdná posloupnost
ϑ	teplota [$^{\circ}\text{C}$]
λ	výstupní funkce konečného automatu, $\lambda: Q \rightarrow Y$
Σ	vstupní abeceda konečného automatu
Σ^{Σ}	množina vstupů z technologického procesu, jako podklad pro Σ

- τ výstupní funkce konečného automatu
 Φ přechodová funkce konečného automatu (*jiné značení*)
 (a, b, c, \dots) symboly abecedy (viz. teorie konečných automatů) *písmena ze začátku abecedy*
 (u, v, w, \dots) slova (viz. teorie konečných automatů) *písmena z konce abecedy*
 \wedge prázdné slovo

Seznam použitých zkratk

Č _{<i>i</i>}	<i>i</i> -té čerpadlo
DCS	Distribuovaný řídicí systém (Distributed Control System)
DFA	Deterministický konečný automat (Deterministic Finite Automaton)
F _{<i>i</i>}	<i>i</i> -té měření průtoku
FSM	Finite State Machine – konečný automat <i>v anglické literatuře</i>
KA	Konečný Automat
L	měření hladiny
M _{<i>i</i>}	<i>i</i> -tý pohon
MIMO	systém s více vstupy a výstupy (Multi Input-Multi Output)
MPC	Model Predictive Control
NFA	nedeterministický konečný automat (Non-deterministic Finite Automaton)
PLC	programovatelný automat (Programmable Logic Controller)
ŘS	řídicí systém
SISO	systémy s jedním vstupem a jedním výstupem (Single Input-Single Output)
V _{<i>i</i>}	<i>i</i> -tý ventil
θ	měření teploty

Seznam obrázků

<i>Obr. 1 Schéma přechodových stavů a činností</i>	16
<i>Obr. 2 Jednorozměrový systém SISO</i>	20
<i>Obr. 3 Vícerozměrový systém MIMO</i>	20
<i>Obr. 4 Prediktivní model</i>	26
<i>Obr. 5 Model a soustava</i>	27
<i>Obr. 6 Základní princip MPC regulátoru</i>	28
<i>Obr. 7 Automat</i>	33
<i>Obr. 8 Zpracování dat z množiny α, měření v čase α_{T_i} a linearizovaný trend</i>	43
<i>Obr. 9 Konečný automat pro prediktivní diagnostiku – tabulka</i>	46
<i>Obr. 10 Konečný automat pro prediktivní diagnostiku – stavový diagram</i>	47
<i>Obr. 11 Konečný automat - Stavový strom</i>	48
<i>Obr. 12 Konečný automat pro prediktivní diagnostiku - stavový strom</i>	48
<i>Obr. 13 Příklad technologického procesu – výroba produktu A</i>	50
<i>Obr. 14 Tabulka přechodových stavů - výroba produktu A</i>	53
<i>Obr. 15 Přechodová tabulka doplněná o přechody do poruchových stavů</i>	55
<i>Obr. 16 Stavový diagram technologického procesu – výroba produktu A</i>	56
<i>Obr. 17 Příklad technologického procesu – výroba produktu maziva mícháním olejů</i>	60
<i>Obr. 18 Tabulka přechodových stavů – výroba maziva</i>	65
<i>Obr. 19 Stavový diagram technologického procesu - výroba maziva</i>	68
<i>Obr. 20 Prediktivní diagnostika technologického procesu – blokové schéma</i>	69
<i>Obr. 21 Časová závislost přechodů a událostí</i>	71
<i>Obr. 22 Časová řada průběhu přechodu $\alpha_{q_0} \rightarrow q_1$, a její linearizace</i>	74
<i>Obr. 23 Časová řada stavu α_{q_1}, a její linearizace</i>	74
<i>Obr. 24 Časová řada průběhu přechodu $\alpha_{q_1} \rightarrow q_2$, a její linearizace</i>	75
<i>Obr. 25 Časová řada stavu α_{q_2}, a její linearizace</i>	75

<i>Obr. 26 Časová řada průběhu přechodu $\alpha_{q2} \rightarrow q_3$, a její linearizace</i>	76
<i>Obr. 27 Časová řada stavu α_{q3}, a její linearizace</i>	76
<i>Obr. 28 Časová řada průběhu přechodu $\alpha_{q3} \rightarrow q_4$, a její linearizace</i>	77
<i>Obr. 29 Časová řada stavu α_{q4}, a její linearizace</i>	77
<i>Obr. 30 Časová řada průběhu přechodu $\alpha_{q4} \rightarrow q_5$, a její linearizace</i>	78
<i>Obr. 31 Tabulka změřených hodnot doby trvání α_{q1}</i>	79
<i>Obr. 32 Časová řada stavu α_{q1} a její linearizace</i>	80
<i>Obr. 33 Tabulka změřených hodnot doby trvání $\alpha_{q1 \rightarrow q2}$</i>	81
<i>Obr. 34 Časová řada stavu $\alpha_{q1 \rightarrow q2}$ a její linearizace</i>	81

1. Úvod

1.1. Formulace problému

Prediktivní diagnostika je v posledních letech často zpracovávanou problematikou a je jedním z principů moderního řízení. Její využití se však soustřeďuje zejména na odhad spotřeby energií a různé prediktivní regulátory např. [1], prediktivní řízení [2], případně pouze pro diagnostiku bez možnosti dalšího využití získaných dat například právě pro predikci. Takovéto využití je však pouze dílčím využitím prediktivní diagnostiky a její nasazení pro predikci chování technologického procesu jako komplexního celku se neprávem opomíjí. Tato práce se zabývá právě využití prediktivní diagnostiky jako nedílné součásti moderního řízení technologického procesu.

1.2. Současný stav problematiky a přínos disertační práce

1.2.1. Úvod

Prediktivní diagnostika patří beze sporu mezi moderní metody řízení technologického procesu. Moderní řízení technologického procesu nepředstavuje pouze řízení samotné technologie, ale zahrnuje i možnosti dalšího využití dat získaných měřeními, případně výpočtem.

Hlavními v současnosti využívanými principy pro moderní řízení technologického procesu jsou zejména:

- Návrh algoritmu optimálního řízení
 - o Ve standardních situacích
 - o V oblastech mimo standardní meze technologického procesu
- Technická diagnostika
 - o Detekce poruch (např. vibrační diagnostika)
 - o Plánování oprav (prediktivní údržba)
- Bezpečnost
 - o Požár
 - o Únik média
 - o Vandalismus

Cílem této práce je rozšířit pohled na moderní řízení, jako na řízení technologického procesu a zejména rozšířit oblast moderního řízení v části technické diagnostiky o prediktivní diagnostiku.

Prediktivní diagnostika není v praxi neznámým pojmem a je v současnosti využívána v případě, kdy chceme plánovat údržbu zařízení (prediktivní údržba), při prediktivní diagnostice točivých strojů, nebo pro odhad spotřeby energií. Dále je pojem prediktivní diagnostika znám i z medicíny kde se využívá k určení, zda a jak bude léčba u pacienta účinná. Zde se však zabývám prediktivní diagnostikou, kdy je možné předvídat chování technologického procesu zejména v případech, kdy technologický proces začne vykazovat nestandardní chování, tedy ne pouze mechanické poruchy způsobené zařízením (porucha motoru, čerpadla, měření, atd.), ale i poruchy vyvolané technologií (nesprávný materiál, nedodržení technologického postupu, atd.). V případě chyby vyvolané pouze technologií budou zařízení, jako jsou akční členy, měření, řídicí automat, apod., vykazovat správné bezporuchové chování. Výsledkem je, že zařízení je sice v pořádku, ale výstup z technologického procesu je nesprávný. Náprava je možná právě tehdy, když se zaměříme na diagnostiku technologického procesu.

Protože pro predikci je velmi důležité, aby data pro vyhodnocení byla správná, je potřeba se při prediktivní diagnostice zabývat i zajištěním důvěryhodnosti měření (kredibilita/diskredibilita senzoru). Zde může pomoci diagnostika pro údržbu, která vyhodnocuje jednotlivá čidla, případně je možné zajistit správnost údajů jiným způsobem jako je například výběr 2 ze 3, zdvojená měření apod.

1.2.1. Současný stav

Při studiu současného stavu dané problematiky jsem se zaměřil zejména na diagnostiku (prediktivní diagnostiku) a konečné automaty. V této kapitole popisují současný stav, diagnostiky a způsob jak jsou využívány konečné automaty pro diagnostiku. Teorie k diagnostice a zejména ke konečným automatům je popsána v samostatných kapitolách a to v kapitole 2. Prediktivní diagnostika a dále v kapitole 3.3.2 Teorie konečných automatů, kde je tato teorie popsána.

1.2.1.1. Diagnostika

Metodika a způsoby prediktivní diagnostiky, které jsou v současné době publikovány, se týkají zejména diagnostiky pro údržbu a diagnostikování chyb zařízení. Diagnostikou pro údržbu se zabývá např. článek [3] kde autor řeší sledování pohonů a diagnostiku v PC na základě údajů získaných monitorovacím systémem. Všechny důležité parametry jsou průběžně sledovány, aby se mohly včas zachytit a odstranit případné poruchové stavy.

V článku [4] se autoři zabývají integrováním systému řízení a monitorování topných kabelů do celkového systému řízení technologického procesu. Tato integrace zahrnuje mnoho výhod jako např. zefektivnění otápění tras, sdílení dat se systémem řízení technologického procesu, sledování hodnot v reálném čase a také možnost diagnostiky otápěných tras, protože porucha otápění může zapříčinit chybu v technologickém procesu, jehož jsou nedílnou součástí. Pro účely diagnostiky se využívá vyhodnocení změřených dat v závislosti na znalosti historických dat.

Autoři publikace [5] se zabývají prediktivní diagnostikou pro zajištění spolehlivosti kabelů. Představují diagnostické programy stárnutí průmyslové izolace kabelů provedené z extrudovaných materiálů¹ (orig. extruded dielectric cable) pro jmenovité napětí 5kV a vyšší. Autoři představují Off-line techniku kdy pomocí PD² diagnostického testu realizují prediktivní diagnostiku využitelnou pro kontrolu izolace kabelů při uvádění do provozu a při periodických kontrolách. Zkoušení kabelů se provádí v souladu se standardem IEEE 400 [6], na který se autoři [5] odvolávají, zkoušení kabelů lze provádět destruktivní nebo nedestruktivní diagnostikou. Účelem nedestruktivní diagnostiky je sledovat podmínky kabelových systémů a určit zda tyto podmínky vyhovují požadovaným průmyslovým standardům. Off-line technika využívá přiměřeně nízké napětí po krátký časový okamžik, aby bylo jisté, že nedojde k poškození kabelu. V některých případech lze využít i on-line PD test, kdy není třeba kabel odpojovat, tento test má však relativně velkou míru nespolehlivosti. Test popisují tak, že na kabel je připojen rezonanční transformátor 50/60Hz, napětí je rychle zvýšeno až na hodnotu U_0 , kde zůstane několik minut a následně je plynule zvýšeno až na hodnotu 2, případně 2,5 U_0 . Na této hodnotě setrvá po dobu přibližně 5 sec a následně je vráceno na nulovou hodnotu, jak nejrychleji je to možné. Během tohoto procesu probíhá sběr dat, která se následně vyhodnocují pro účely nedestruktivní diagnostiky.

¹ Silové kabely s pevnou, výtlačně lisovanou izolací.

² PD – orig. Partial Discharge. Elektrický výboj, který nepřeklene zcela prostor mezi dvěma elektrodami

V publikaci [7] se autoři zabývají on-line prediktivní diagnostikou vzniku poruchových stavů budícího vinutí a rotačního usměrňovače bezkartáčových synchronních generátorů. Predikce a vzájemné působení těchto poruch vychází z analýzy teorie bezkartáčových generátorů.

Prediktivní diagnostikou pro elektromechanický systém se zabývají autoři [8]. Jedná se opět o diagnostiku pro údržbu, tzv. CBM (Condition Based Maintenance) údržba, jejíž základní myšlenkou je, že údržba stroje, nebo zařízení se provádí pouze tehdy, když existují objektivní důkazy o nutnosti údržby, nebo hrozícího selhání. Aby toto bylo možné, CBM nutně zahrnuje prediktivní diagnostiku. Stav stroje se zjišťuje pomocí senzorů, které měří hodnoty, jako jsou vibrace, teplota, tlak, přítomnost nečistot oleje apod. Tato multisenzorová měření sama o sobě nezlepšují možnosti diagnostiky, musí být vybírána taková data, která jsou relevantní k dané úloze. Získaná data jsou dále zpracovávána pomocí specifických funkcí k identifikaci daných poruch. Multisenzorové systémy (orig. Multisensor Data Fusion), jsou velmi často používány vzhledem k tomu, že každý senzor může přispět jedinečnou množinou svých získaných dat k diagnostice zařízení. Diagnostiku realizují klasifikací (klasifikace, jako jedna z možností predikce) a pro rozhodování popisují existující techniky, jmenovitě výběr (orig. Voting) kdy je výběr, které výstupy z multisenzorových měření budou zasahovat do predikce, prováděn většinovým způsobem. Variací je vážený výběr (orig. weighted voting) ve kterém mají čidla přiřazenu relativní váhu pro výběr. Další metodou je vážený výběr (orig. weighted decision fusion) kdy rozhodování je realizováno kombinací výstupů z různých snímačů. Snímače mají přiřazenu váhu dle své přesnosti a spolehlivosti. V případě, že tyto údaje nejsou známy, se z váženého výběru stane vážená volba (viz výše). Další možností je Bayesovská statistika (orig. Bayesian Interference) blíže [9], která poskytuje metodiku pro výpočet pravděpodobnosti a dovoluje zpřesňovat výchozí pravděpodobnost na základě následných relevantních skutečností. Autoři se dále zabývají diagnostickými modely vhodnými pro CBM. Diagnostika, na základě modelu může úspěšně detekovat závadu, nebo posuzovat stav zařízení protože získané matematické modely jsou ověřovány v mnoha podmínkách a v různých stavech, modely pak pracují na základě znalostní databáze. Pro diagnostické modely je podstatné aby postihovaly charakteristické účinky poruch a jejich vztah k pozorovaným údajům. V další části se autoři věnují příkladům.

V článku [10] autoři popisují integrovaný systém pro podporu rozhodování pro diagnostiku poruch a efektivní plánování údržby. Tento Integrovaný systém se skládá ze tří

spolupracujících modulů, z Diagnostického modulu, Modulu plánování údržby a Plánovacího modulu. Popisovaný systém pomáhá při rozhodování které informace a o kterých stavech procesu se využijí pro diagnostiku.

Predikčními modely a metodami které lze využít při řešení problematiky pravděpodobnosti vzniku události na základě expertního pravidlového systému je uvedeno v publikaci [11]. Pro modelování jsou zde navrženy Fuzzy expertní systémy.

Hybridní systémy, které odstraňují problémy při využívání neuronových sítí pro diagnostiku, se zabývá článek [12]. Neuronové sítě mohou být nedokonale natrénovány a potom se mohou vyskytnout i chyby při diagnostice. Naproti tomu hybridní inteligentní systémy jsou pro diagnostikování chyb vysoce spolehlivé. Přímou pro diagnostiku byly vyvinuty různé neuronové sítě, tyto však mají dvě hlavní nevýhody: nejsou schopny rozlišit různé poruchy, které mají stejné příznaky a znalosti, které jsou k dispozici pro trénování sítě, mohou být neúplné. Proto nejsou standardní neuronové sítě příliš vhodné pro diagnostiku. Architektura hybridního inteligentního systému se skládá ze zpětné neuronové sítě nazývané generátor hypotéz (orig. hypothesis generator) a ze zpětné neuronové sítě nazývané Systémový simulátor (orig. system simulator). Tyto dvě neuronové sítě jsou integrovány s rozhodovacím procedurálním algoritmem. Při startu chybové diagnostiky je generátor hypotéz založen na reálném vzoru a generuje odhad (hypotézy) chybových událostí. Systémový simulátor je potom realizován podle vzoru obdržených hypotéz a hypotetických příznaků. Rozhodovací algoritmus porovná hypotetické příznaky s reálnými pro validaci hypotetických příznaků. Pokud nejsou hypotetická data platná, rozhodovací algoritmus je zreviduje a podle těchto nových hodnot nastaví Systémový simulátor pro další validaci. Pokud rozhodovací algoritmus rozhodne, že hypotetická data jsou platná, informuje obsluhu o možné chybě s žádostí o potvrzení. Obsluha informuje systém o výsledku potvrzení. Systém se nastaví podle revidovaného vzoru.

Otázkou prediktivní údržby se přehledně, z hlediska organizačního a technicko-ekonomického zabývá publikace [13]. V publikaci jsou popisovány i různé techniky při prediktivní údržbě.

V prediktivní údržbě a diagnostice, existují různá firemní řešení chráněná patenty např. [14], [15] a [16]. Patent [14], firmy Westinghouse se zabývá podnikovou údržbou

pomocí prediktivní diagnostiky. Patent se týká oblasti řízení informačních systémů a zejména integrovaného monitorovacího a diagnostického systému využitelného nejen pro přímé řízení technologického provozu, ale i pro údržbu a technická oddělení jaderných elektráren. Integrovaný informační systém, který řídí výrobu, sbírá provozní data, která jsou využívána k monitorování a řízení provozu, tato data lze také využít i pro prediktivní údržbu, pro kterou lze zajistit ještě další doplňující měření určená pouze pro prediktivní diagnostiku. Patent společnosti Gtech Corporation [15] se týká prediktivní diagnostiky a řízením poruch herních systémů, kdy je každý herní automat vybaven senzory, které monitorují a zaznamenávají jeho chování, problémy nebo události a další diagnostická data. Jednotlivým událostem jsou přiřazeny priority. V řídicím centru má obsluha přehled o aktuálním stavu všech servisovaných hracích zařízení, které mohou být ve třech stavech „green“ zařízení je v pořádku a funkční, „red“ zařízení je v poruše a čeká na opravu a „amber“ což je stav mezi „green“ a „red“. Na základě uvedené vizualizace je následně řízena a plánována údržba. Prediktivní diagnostikou pomocí programovatelného automatu se zabývá patent firmy Schneider Automation [16], týkající se diagnostického systému, který zpracovává údaje z technologického procesu programovatelným automatem. Patent navrhuje diagnostický systém, který lze zabudovat do programovatelného automatu (PLC) a který je navržen tak, aby mohl predikovat i proces, pro který nelze vytvořit matematický model. Toho je dosaženo pozorováním jednoho, nebo více jednoduchých procesních signálů, které jsou použity k učení modelu a které mohou být využity pro predikci výskytu poruchové události, nebo závady v procesu. Část sloužící k pozorování je přímo zakomponována v PLC, část sloužící k učení, která bude využívat detekční algoritmus, může být také zahrnuta v PLC, nebo při nedostatku místa v jiném zařízení schopném zpracovávat data a komunikovat s PLC.

Jak vyplývá, z výše uvedené literatury, pro diagnostiku je potřebné zjišťovat a archivovat relevantní údaje, které je nutno v dalším kroku dále zpracovávat, vyhodnocovat a porovnávat s aktuálními údaji. Ve výše citovaných publikacích se autoři zabývají prediktivní diagnostikou zejména pro diagnostiku poruch pro účely údržby. Technologickým procesem se zabývá hlavně literatura [12] a [16]. Pro vyhodnocení jsou zde používány modely, učící algoritmy případně neuronové sítě. Na rozdíl od těchto řešení se ve své práci zaměřuji na vyhodnocování a diagnostiku pomocí konečných automatů.

1.2.1.1. Konečné automaty

Využívání konečných automatů rozličnými způsoby pro predikci se v literatuře vyskytuje již delší dobu, např. v článku [17] se autoři zabývají využitím konečného automatu (FSM) pro určení vývoje populace, jako míry náhodnosti dané binární sekvence. Na FSM s binární vstupní i výstupní abecedou pohlíží jako na prediktor binární sekvence. Používání náhodné binární sekvence (Random Binary Sequence) je běžné v aplikované matematice (integrace, kryptografie, statistika) a její využívání se nadále rozvíjí. Autoři článku navrhují „evoluční algoritmus“ (evolutionary algorithm), který je schopen předvídat hodnoty v binární řadě lépe než náhodně. Základní myšlenkou je využití „evolučních“ schopností predikce navrhovaného algoritmu k omezení náhodnosti v sekvenci. Pro odhalení zákonitostí v sekvenci používají konečný automat. V tomto směru se zkoumáním binárních sekvencí zabýval M.O'Connor [18] a zobecněním jeho práce na libovolné sekvence se v případě Mealyho automatu zabývají ve své studii Broglio a Liardet [19], kde dokazují, že existuje efektivní proces k určení hodnot v náhodných sekvencích a to právě pomocí deterministických konečných automatů.

V [19] se autoři zabývají situací, kdy máme nekonečnou řadu čísel a snažíme se odhadnout číslo v každém následujícím kroku. Autoři toto řeší pomocí prediktoru, který je definován na konečné řadě čísel. Ve své práci se zaměřují na prediktory na principu konečného automatu. Autoři studují možnosti rozpoznání, nebo generování náhodné sekvence deterministickým konečným automatem. V práci [19] je uveden důkaz, že pokud máme konečnou množinu A o q číslicích, kde $q \geq 2$, a nekonečnou sekvenci $\mathbf{u} := (u_n)_{n \geq 0}$ číslic u_n (\mathbf{u} je také označováno jako nekonečné slovo). A pokud je \mathbf{u} náhodné, pak lze automatem po postupném přečtení u_0, \dots, u_{n-1} stanovit hodnotu u_n s pravděpodobností větší než $1/q$. Z těchto závěrů vychází i článek [17].

V pracech [17] a [19] je uvedena definice prediktoru P , jako $P : A^* \rightarrow A$. Vzhledem k jakémukoliv slovu $w \in A^*$ říkáme, že P predikuje symbol $a \in A$ pokud zná slovo w , pokud $P(w)=a$.

Mějme $w = w_1 w_2 \dots w_n$, a řadu symbolů $z A$ o délce n a množinu $w[k] = w_1 w_2 \dots w_{k-1}$, pro $k \in \{1, 2, 3, \dots, n+1\}$ a kde $w[1]$ je prázdné slovo, značené Λ .

Na základě výše uvedeného je potom jednoduché zkonstruovat prediktor P , schopný dokonale predikovat slovo w . Je zcela postačující vybrat $P(\wedge) = w_l$ a potom platí pro všechna $i \in \{1, 2, \dots, n\}$, $P(w[i]) = w_i$.

S pomocí libovolného konečného automatu M (Mealyho konečného automatu) lze přiřadit automatický prediktor P_M , definovaný na tomto automatu.

$$\forall w \in A^*, P_M(w) = \tau \circ \Phi_{w_n} \circ \Phi_{w_{n-1}} \circ \dots \circ \Phi_{w_1}(s_0), \quad \text{kde } P(\wedge) = \tau(s_0).$$

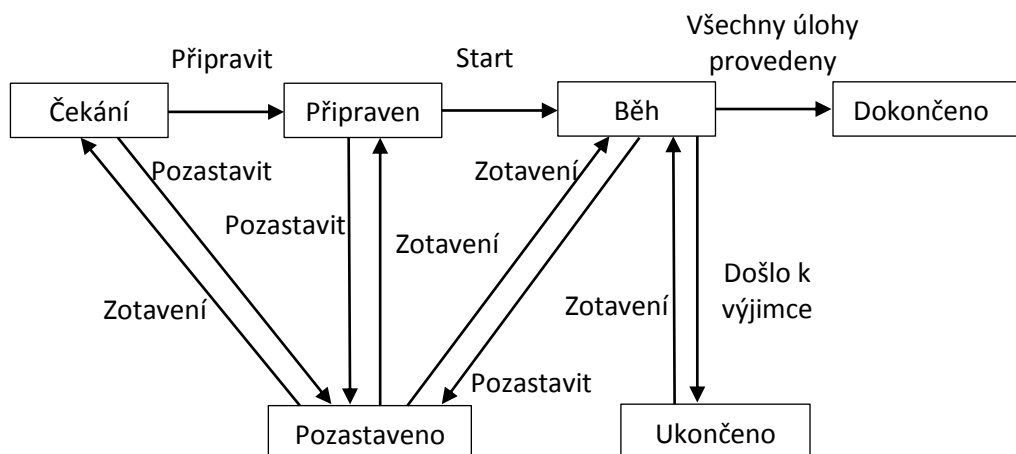
Více informací k teorii konečných automatů je uvedeno v kapitole 3.3.2 Teorie konečných automatů.

Problematikou predikčních algoritmů v číselných řadách, založených na konečných automatech, se zabývá také práce [20]. V této práci je uveden prediktor P , definovaný na konečném automatu α s konečným počtem stavů. Automat má vstup i výstup. Nejprve je spuštěn konečný automat, a pokud je vstupem symbol $a \in A$ a i_0 je počátečním stavem, získáme po nějakém čase symboly $\{P(a); a \in A\}$, kdy $x_1 x_2 \dots x_n$ naplní v po sobě jdoucí řadě vstupy. Po x_n tedy máme $\{P(a/x_1 x_2 \dots x_n); a \in A\}$.

Okrajově se zde zmiňují i o práci [22], kde se autoři zabývají predikcí na základě jednoduchých rekurentních sítí (simply recurrent network SRN), využívajících strukturu aktivace přes množinu veličin od kroku v čase $t-1$, společně s prvem v čase t , za účelem predikce elementu v čase $t+1$. Pokud je síť trénovaná na řetězci s konečnou množinou stavů (orig. finite state grammar), může se stát dokonalým prediktorem pro uvedenou množinu stavů (gramatiku). Autoři [22] se opírají o práci J.F. Elmana [21], který prezentoval, jednoduchou rekurentní síť (SRN), která má potenciál zvládnout nekonečný soubor sekvencí s omezenými prostředky k proceduře učení, probíhající v *lokálním* čase. V [22] je prezentováno, že SRN se může naučit napodobovat konečný automat a to jak v jeho chování, tak jeho reprezentací. Síť je tedy možné trénovat pomocí konečného automatu.

V literatuře [23] se autoři zabývají teorií konečných automatů při modelování softwarového procesu z pohledu paralelních časových automatů. Základem je rozšíření konečných automatů na paralelní automaty, které realizují plánovanou činnost, přidělování zdrojů a řízení procesu. Autoři představují konstrukci softwarového procesního modelu

založeného na časových paralelních automatech. Na *Obr. 1*, převzatém z [23], jsou znázorněny činnosti, stavy, jejich přechody a vzájemné vazby. Činnosti korespondují s množinou uzlů N (orig. node) časových paralelních automatů. Množina uzlů reprezentuje soubor veškerých aktivit celého procesu, Každý stav S (orig. state) představuje soubor činností, které by se mohly objevit kdykoliv a v jakémkoliv dynamickém stavu.



Obr. 1 Schéma přechodových stavů a činností

Práce [23] se dále zabývá sekvenčními strukturami a smyčkami procesu. Vychází ze skutečnosti, že změna stavu v automatu nastane, zatímco jsou čteny znaky v časovém paralelním automatu a dále že, události a podmínky aktivují změny činností v softwarovém procesu. Tedy události a podmínky mohou být vstupními znaky do časového paralelního automatu. Množinu událostí označují autoři jako Events a množinu podmínek jako Condition, na základě tohoto vyjádření definují, že vstupní data do časového paralelního automatu lze vyjádřit jako $\Sigma = \text{Events} \times \text{Conditions}$.

1.2.2. Přínos disertační práce

V literatuře [17], [19], [20] a [22] se autoři zabývají predikcí po sobě jdoucích prvků v číselných řadách $(t-n, \dots, t-1, t, t+1, \dots)$ pomocí konečných automatů. Publikace se zabývají

zejména oblastí aplikované matematiky, jako je integrace, kryptografie a statistika. Ve své práci vycházím ze skutečnosti, že i v technologickém procesu lze stanovit časové řady, jejichž prvky jsou závislé na chování a stavu technologického procesu a pomocí predikce lze diagnostikovat chování technologického procesu.

V literatuře [23] se autoři zabývají vytvořením procesního modelu na základě konečných automatů. Zmiňují metodu časových paralelních automatů jako modelů softwarového procesu, založenou na základě konečných automatů pro strukturu softwarových procesů, která může být nápomocna vývojářům k rozvržení zdrojů, analyzování a řízení procesu. Na rozdíl od této práce se zaměřuji na vytvoření takové struktury konečného automatu, který lze využít pro modelování technologického procesu, tedy nezohledňuji pouze softwarové struktury, ale zaměřuji se na technologický proces jako celek. Takto vytvořený konečný automat následně představuji jako nástroj pro diagnostikování technologického procesu.

Hlavním přínosem této práce je nastavení nového pohledu na teorie moderního řízení, oproti současnému převládajícímu názoru zdůrazňuji skutečnost, že moderním řízením nejsou pouze sofistikované regulace, ale že je zde ještě velká řada možností, které stávající teorie automatizace nabízí a které se neprávem opomíjejí. Pouze je potřeba se na tyto systémy podívat z jiného úhlu pohledu. Hlavním tématem mé práce je prediktivní diagnostika technologického procesu. Na téma predikce je zpracováno velké množství odborné literatury, její studium bylo důležitým podkladem pro tuto práci a je uvedena v seznamu literatury. Z nabízených možností pro predikci jsem se zaměřil na predikci pomocí modelů, která se jeví jako velmi vhodná pro predikci technologického procesu, protože se nezaměřuje pouze na predikci určité veličiny, ale pomocí modelu lze zohlednit celý technologický proces komplexně, tedy včetně všech vnějších návazností a vnitřních vazeb, které technologický proces ovlivňují.

V disertační práci navržený prediktivní model pro diagnostiku vychází z teorie konečných automatů, neboť technologický proces a konečný automat vykazují obdobné chování. Využití konečného automatu, který byl původně navržen pro využití v informatice, jako modelu technologického procesu je právě jedna z takových nových možností, jak využít stávající teorii řízení novým způsobem. Konečný automat jako model technologického procesu je možné použít prakticky beze změny, pro účely prediktivní diagnostiky je však

potřeba obecně definovaný konečný automat doplnit a rozšířit. V disertační práci definuji takovýto konečný automat pro prediktivní diagnostiku, jehož vstupy jsou události mající vliv na technologický proces (události v technologickém procesu) a který využívá, zejména, časových údajů o trvání stavů a přechodů v technologickém procesu k diagnostice chování technologického procesu v čase. Na základě těchto údajů lze odhalit blížící se fatální stav, případně plánovat servisní zásahy. Využitelnost této teorie v praxi je demonstrována na příkladech uvedených v kapitole 3.4.

1.3. Cíle disertační práce

Původním impulsem k napsání této práce bylo zaměření na moderní metody řízení technologického procesu. Moderním řízením se totiž v současné literatuře obvykle myslí pouze vyspělé metody regulace, kdy se vždy jedná o rozličné metody spojitého řízení (stavová regulace, robustní prediktivní regulátory...) a nezabývá se řízením diskretních soustav, tedy řízením událostí což je zřejmé i z literatury, např. článek s následnou polemikou v odborném časopise [24], nebo v rámci odborných přednášek [25]. Pokud si představíme technologický proces jako soubor sekvencí a regulací a vyjdeme-li z předchozího tvrzení, že moderními metodami se v současné době myslí především regulace, je patrné, že pro řízení technologického procesu si nevystačíme pouze s těmito vyspělými metodami regulace, které představují spojitou část technologického procesu, ale je nutné se zabývat i diskretními částmi technologického procesu. Diskretní část představují povely, sekvence, stavy, případně receptury apod., které jsou podstatné pro průběh vlastního technologického procesu. Diskretní data z technologického procesu vypovídají o stavu a chování tohoto procesu, je možné je dále využít k diagnostice a použitím vhodných metod i k prediktivní diagnostice technologického procesu jako celku.

Cílem disertační práce je představit celkový pohled na řídicí algoritmy a technickou diagnostiku technologických procesů. V práci je představena možnost využití dat získaných z technologického procesu. Data, která lze získat z technologického procesu jsou dvojitá. Jsou to **technologická data** tedy data přímo se na technologickém procesu podílející (např. teplota média, hladina média, poloha ventilu,...) a **netechnologická data**, která se na vlastním procesu přímo nepodílejí, ale jsou jeho nedílnou součástí, ať již chtěnou či nechtěnou (např. teplota vinutí motoru, vibrace, doba trvání procesu, doba přeběhu armatury,...). Součástí disertace je popis možností využití dat získaných z technologického procesu pro diagnostické

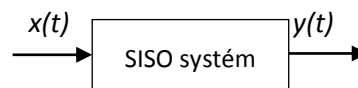
účely tohoto procesu, zejména pro prediktivní diagnostiku a dále je zde položen základ pro posunutí možností využití moderních metod z oblasti teorie automatizace a diagnostiky směrem k řízení technologického procesu jako kompaktního celku.

2. Prediktivní diagnostika

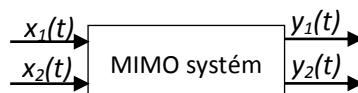
2.1. Základní pojmy

Základní pojmy, se kterými budu dále pracovat, jsou uvedeny v různé literatuře, zahraniční literatura je citována v odstavci 1.2.1.1 Diagnostika [3] až [16], česky pak např. [26], [27] přehledně pak v článku [28]. Mezi základní pojmy v predikci patří výrazy **vysvětlovaná veličina** a **vysvětlující veličina** (nazývaná také prediktor). Vysvětlovaná veličina je veličina, jejíž vývoj sledujeme a kterou chceme predikovat, vysvětlující veličinou je potom veličina, případně veličiny, na nichž je vysvětlovaná veličina závislá. Pro $y = f(x)$, kde f vyjadřuje funkční závislost je tedy y vysvětlovanou veličinou, a x vysvětlující veličinou.

Systémy s jedním vstupem a jedním výstupem nazýváme jednorozměrovým, zkráceně **SISO systém** z anglického Single Input-Single Output. Systém s více vstupy a výstupy nazýváme vícerozměrovými, zkráceně **MIMO systém** z anglického Multi Input-Multi Output. Oba systémy jsou uvedeny na *Obr. 2* resp. na *Obr. 3*



Obr. 2 Jednorozměrový systém SISO



Obr. 3 Vícerozměrový systém MIMO

2.2. Predikce

Predikci, tedy předpovídání budoucího stavu, nebo chování lze v oboru řízení a průmyslové automatizace rozdělit do dvou hlavních směrů:

- predikce pro řízení, kdy hledáme optimální zásah do technologického procesu pomocí různých prediktivních regulátorů,
- predikce pro diagnostiku, kdy sledujeme veličiny/proces a predikujeme jejich budoucí vývoj, či směr vývoje a zároveň vyhodnocujeme, zda sledovaná (é) veličina (y) /proces budou mít negativní, nebo nějaký dopad na technologický proces, a čím je toto chování způsobeno.

V dalším textu se zabývám výhradně diagnostikou pro predikci. Avšak obecné principy z oblasti predikce, základní pojmy, ale i metodika predikce je obecně platná a dotýká se i prediktivní diagnostiky technologického procesu.

Prediktivní diagnostika může pro svou potřebu využívat široký matematický aparát a různé diagnostické metody [28]. Podstatou predikce je však dostatek důvěryhodných informací (kredibilita/diskredibilita měření) o současném stavu technologického procesu a o stavu ostatních (netechnologických) podmínek které mají vliv na proces. Dále je důležitá znalost vlastního technologického procesu a v neposlední řadě jsou pro diagnostiku důležitá i historická data. Tato skutečnost je s větším, či menším důrazem zmiňována i v literatuře [3] až [16] zabývající se diagnostikou, respektive prediktivní diagnostikou, že totiž pro diagnostiku (prediktivní diagnostiku) je vždy důležitý sběr a archivace dat a jejich následné vyhodnocení.

V technologickém procesu se vyskytují jak veličiny spojité, tak i veličiny diskrétní a z toho důvodu je třeba pohlížet na systém pro vyhodnocení technologického procesu, tedy i pro diagnostiku, jako na systém hybridní tj. na systém zpracovávající jak spojité tak i diskrétní hodnoty.

Prediktivní diagnostiku můžeme rozdělit do dvou základních skupin a to na prediktivní diagnostiku s modelem a prediktivní diagnostiku s využitím numerických a statistických metod.

2.3. Prediktivní analýza

Prediktivní analýza (Predictive Analytics) [29] je široká oblast zahrnující statistické a analytické metody pro vytváření modelů, které předvídají budoucí události, nebo chování. V principu jde o získávání informací z dat a o jejich dalším využití pro předvídání budoucího vývoje, jedná se zejména o statistiku, strojové učení a dolování data (data mining). Prediktivní analytika je využívána hlavně v pojišťovnictví, finančních službách, marketingu, zdravotnictví, medicíně a telekomunikacích. V těchto odvětvích se pomocí prediktivní analytiky zkoumá vývoj prodeje, chování zákazníků, odhalování finančních podvodů, určení schopnosti splácet úvěr apod. Prediktivní analýza se také využívá pro prediktivní údržbu ke zjištění stavu zařízení.

2.4. Prediktivní diagnostika pro údržbu

Prediktivní diagnostika pro údržbu je v současné době nejrozšířenější prediktivní diagnostika používaná v průmyslu, existuje mnoho odborných publikací, ať již vydaných v tištěné formě [13], [30], nebo v rámci odborných periodik na internetu např. www.plantengineering.com, případně česká mutace www.udrzbapodniku.cz, které se věnují problematice řízení podniku. Prediktivní diagnostika pro údržbu, využívá metod prediktivní analýzy pro včasné varování před možnou hrozbou selhání zařízení a procesů, které by byly jinak nezjistitelné. Prediktivní diagnostika je založena na porovnávání dat se shromážděnými a analyzovanými údaji ze stovek miliónů strojních hodin a několika desítek tisíc událostí různých typů zařízení. Tato data pocházejí ze světové databanky informací, do které byla data shromažďována 10 let (viz [30]). Hlavním problémem se zde jeví, že máme, sice uspořádané, ale rozsahem velké množství dat a údajů z událostí, které již nemusí nikdy nastat a naopak je reálně možný výskyt události v takovéto databázi nepodchycené. Prediktivní diagnostika pro údržbu je účelná pro jednotlivá zařízení (pohony, čerpadla, měření,...), kde pomocí diagnostiky můžeme zajistit prodloužení životnosti zařízení, zvýšit jeho spolehlivost a snížit náklady na servis a údržbu. Diagnostiku vlastního technologického procesu to však neřeší, je ale možné, a také účelné, využít závěry diagnostiky pro údržbu jako míru důvěryhodnosti jednotlivých zařízení, účastnících se technologického procesu (kredibilita/diskredibilita senzoru).

3. Prediktivní diagnostika technologických procesů

3.1. Metody predikce spojitéch veličin

Různé metody predikce spojitéch veličin, kterými se hlavně v reálném technologickém procesu zabýváme, jsou souhrnně uvedeny v článku [31] jedná se o tyto metody:

a. Extrapoláčnı metody, mají svůj základ v metodách interpolace a aproximace funkcı, kterými se zabývají tradiční numerické metody. Úkolem *polynomiální interpolace* je proložit souborem $n + 1$ bodů (x_i, y_i) , $i = 0, 1, \dots, n$, polynomiální funkci $y = P_n(x)$ stupně nejvýše n , tak, aby procházela všemi zadanými (uzlovými) body. Naproti tomu úkolem *polynomiální aproximace* je nalezení polynomu, který zadanou funkční závislost co nejpřesněji nahrazuje. Jako kritérium kvality aproximace se nejčastěji používá metoda nejmenších čtverců. Obvyklé metody interpolace i aproximace předpokládají ekvidistantní rozdělení uzlových bodů, kdy vzdálenost bodů ve směru nezávislé proměnné (vodorovné osy x nebo t) je konstantní. Existují i jiné metody interpolace a aproximace, kdy jsou místo polynomů (mocninných funkcı) používány jiné typy funkcı, např. racionálně lomené, exponenciální nebo trigonometrické. Získaná funkce nahrazuje ideální závislost uvnitř definičního intervalu, vymezeného dvojicí krajních bodů ze souboru zadaných. Při extrapoláčnıch metodách se interpolující či aproximující funkce prodlouží za hranice svého definičního intervalu (tedy za své krajní uzlové body). Pomocı metod extrapolace funkce je potom možné odhadnout funkční hodnoty pro tyto body, tedy pro $n + 1, n + 2, \dots$ atd. Při predikci se zajímáme o hodnoty sledované funkce v čase a proto je v definičních vzorcích pro interpolaci a aproximaci nezávislou proměnnou čas. Potom je tedy možné extrapolací funkce $y = f(t)$ odhadnout její hodnoty v časových okamžicích $t + 1; t + 2; \dots t + k$. Tedy predikovat budoucí hodnoty funkce.

b. Lineární predikce vychází z předpokladu, že hodnotu sledované veličiny v následujícím kroku můžeme určit lineární kombinací předchozích p vzorků, viz (3.1), kde $s(n)$ je hodnota sledované veličiny v příštím kroku, p je řád lineárního prediktoru a a_i jsou koeficienty predikčního filtru. Pro známé koeficienty je algoritmus výpočetně nenáročný, jedná se o zvláštnı případ systému (filtru) s konečnou impulsnı odezvou.

$$s(n) = \sum_{i=1}^p a_i s(n-i) \quad (3.1)$$

- c. Lineární diferenční predikce určuje hodnotu sledované veličiny pro příští výpočetní krok jako lineární extrapolaci kroku současného a minulého (první diference zleva). Principiálně se jedná o linearizaci sledované veličiny v krátkém časovém úseku. Je to zvláštní případ lineární predikce, při které se jedná o jednoduchou linearizaci sledované závislosti a prodloužení jejího dosavadního vývoje po přímce spojující dva poslední změřené body (sečna grafu ideální funkční závislosti).
- d. Polynomiální diferenční predikce je zobecněním metody lineární diferenční predikce. V každém kroku se provede adaptace, při které jsou z aktuální hodnoty sledované veličiny a z posloupnosti jejích historických vzorků postupně spočteny diference prvního, druhého a dalších řádů (podle řádu polynomiálního prediktoru). Při prvním výpočtu v prediktivní fázi je počítána hodnota pro první predikční krok $t + 1$ tak, že z poslední diference je spočtena aktualizovaná hodnota předposlední diference, z ní se opět spočte diference dalšího nižšího řádu, a dále až se z druhé diference spočte aktualizovaná první diference, ze které se spočítá hodnota sledované veličiny pro první krok. Pro druhý a další predikční kroky je postup stejný.
- e. Statistické metody predikce vycházejí ze skutečnosti, že soustava je někdy interpretována jako náhodný proces, jehož budoucí vývoj lze předvídat na základě znalostí statistických charakteristik a statistických metod.
- f. Neuronové sítě se využívají pro predikci zejména tehdy, když neznáme podstatu mechanismů fungování procesu, který chceme sledovat. Vstupem do takového prediktoru může být aktuální hodnota sledované veličiny a posloupnost minulých vzorků. Pro každý krok predikce je realizována jiná neuronová síť. Pro fungování neuronové sítě je nutné nejprve její natrénování vhodným způsobem na odpovídajících vzorcích dat.
- g. Regresní analýza se využívá pro nalezení modelu, který bude odpovídat závislosti jedné veličiny na ostatních. Jedná se o nalezení analytického vztahu na základě zadaných (nebo

změřených) hodnot sledované veličiny, který co nejpřesněji popisuje její funkční závislost. Takovýmto regresním modelem můžeme potom predikovat hodnoty sledované veličiny. V regresní analýze lze využít různé funkční závislosti. Lineární regrese využívá funkce lineární v parametrech, nelineární regrese zase využívá funkce v parametrech nelineární (např. exponenciální či trigonometrické funkce). Jednoduchá regrese studuje závislost dvou veličin, vícenásobná regrese studuje závislost jedné veličiny na dvou nebo více veličinách.

- h. Modely** se pro predikci používají tehdy, když je lze pro zkoumanou soustavu vytvořit dostatečně věrně, například některou z výše uvedených metod. Takto vytvořený model je možné využít pro předvídání (predikci) chování dané soustavy (nebo i technologického procesu) v čase. To znamená, že model se chová jako prediktor. Model může být představován např. matematickým vztahem, přenosovou funkcí v Laplaceově transformaci, soustavou diferenčních rovnic, programem v simulačním SW (Matlab/Simulink), pravidlovým systémem apod. Model v režimu Off Line (kdy pracuje odděleně a nezávisle na momentálním chování své předlohy) lze využít pro ověření chování modelované soustavy, pro její testování a v případě, že model představuje nějaké komplexnější (např. technologické) zařízení i jako trenažer pro zaškolení obsluhy a nácvik chování v nestandardních situacích. V těchto případech mívá model stejně rychlou odezvu, jako reálná soustava (čas v modelu je shodný s reálným časem) a je žádoucí, aby měl i stejné chování. Model může také pracovat v On Line režimu, kdy je přímo spojen se skutečnou soustavou (svým vzorem), vstupem do modelu jsou reálné hodnoty vstupující do soustavy, ve formě digitalizovaných hodnot. Pokud je model nastaven tak, aby pracoval ve zrychleném režimu oproti reálné soustavě (čas v modelu je rychlejší, než reálný), lze stanovit v předstihu budoucí hodnoty sledované veličiny. Model může být i nedílnou součástí regulačního, nebo diagnostického algoritmu. Kvalita predikce je vždy závislá na tom s jakou přesností lze model pro danou soustavu vytvořit. Modely můžeme rozlišovat podle způsobu, jakým se na pozorovanou soustavu díváme a jaké máme možnosti pro získání popisu této soustavy. Model můžeme vytvořit na základě vnitřního popisu, kdy jsou po matematicko-fyzikální analýze odvozeny vztahy mezi vstupy a výstupem (sledovanou veličinou). Tyto odvozené vztahy jsou potom platné pro celý pracovní rozsah soustavy. Dále můžeme model vytvořit na základě vnějšího popisu, takovýto způsob vytvoření modelu je obvykle používán v případě, kdy je systém příliš složitý na to, aby mohly být získány parametry systému pro jeho úplný matematicko-fyzikální popis. Při

tomto postupu přistupujeme k systému jako k černé skřínce a na základě změřených vstupů a výstupů systému jsou nalezeny jeho parametry. Výhodou této metody je, že není potřeba znát fyzikální princip soustavy, nevýhodou je, že takto získaný model soustavy není platný v celém jejím pracovním rozsahu, ale pouze v rozsahu těch vstupů a výstupů, pro které byl model získán.

3.2. Modely

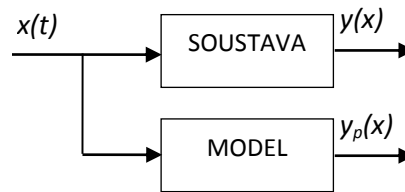
Většina z metod uvedených v odstavci 3.1 je vhodná pro predikování průběhu určité veličiny (a až e). Pro predikování technologického procesu jako celku jsou vhodné statistické metody, neuronové sítě a zejména pak metody s modelem, kterými se budu dále zabývat. Tyto metody lze samozřejmě použít i pro predikci průběhu veličiny.

Je-li znám způsob, jakým bude soustava zpracovávat hodnoty vstupní veličiny $x = f(t)$ na hodnoty výstupní veličiny $y = f(x)$, pak můžeme pro potřeby predikce soustavu nahradit modelem jejího chování se shodným vstupem x a predikovaným výstupem, který označíme y_p . viz Obr. 4.



Obr. 4 Prediktivní model

Takto lze, ze známé hodnoty vstupní veličiny $x = f(t)$, a jejího vývoje, odhadovat (predikovat) budoucí hodnoty výstupní veličiny $y_p = f(x)$. Pomocí modelu můžeme realizovat predikci v režimu off-line, kdy hodnoty vstupu $x = f(t)$ simulujeme, nebo v režimu on-line, kdy model soustavy je připojen paralelně k reálné soustavě viz Obr. 5.



Obr. 5 Model a soustava

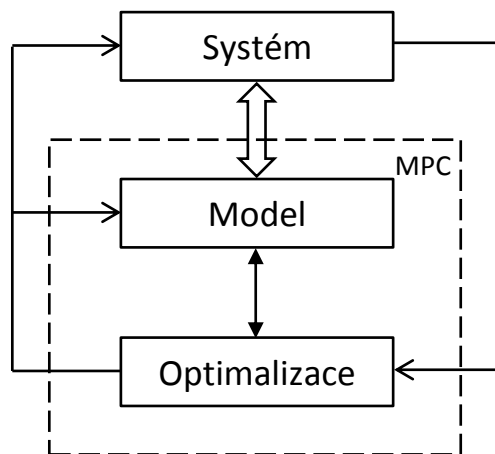
Pomocí on-line modelu, podle *Obr. 5*, můžeme sledováním a vzájemným porovnáváním vývoje a odchylek měřené a vypočtené (predikované) hodnoty realizovat diagnostiku zařízení. Pro účely prediktivní diagnostiky nás nezajímá ani tak přesnost predikované hodnoty (veličiny, nebo stavu zařízení) jako spíš její celkový průběh. Není tedy chybou, pokud $y(x) - y_p(x) \neq 0$, predikovaná hodnota se musí pohybovat v předem definovaných mezích a dále musí být v souladu se svými historickými hodnotami. Pokud tomu tak není je zřejmé, že v soustavě došlo ke změně, tedy reálný proces přestal odpovídat svému modelu, tzn., v soustavě došlo k nechtěné změně. (porucha, chybná manipulace, porucha čidla...).

Model soustavy má často formu matematického analytického vztahu. U dynamických soustav je to obvykle soustava diferenciálních rovnic nebo schéma modelu v některém simulačně zaměřeném výpočetním systému (obvykle Matlab / Simulink). Pro lineární (linearizované) soustavy je obvyklá přenosová funkce v Laplaceově transformaci. Vzhledem k tomu, že v praktických aplikacích jsou algoritmy modelů, řízení i diagnostiky většinou realizovány programem v řídicím systému (PLC, DCS), je přirozené je realizovat jako diskrétní systémy. Pro ně je pak zřejmým prostředkem pro popis soustava diferenčních rovnic a pro lineární systémy i přenosová funkce v transformaci Z, obvykle s proměnnou z . Po přepisu do podoby s proměnnou $z - 1$ získáme formu vhodnou pro vytvoření schématu některé z kanonických forem realizace. To je již přímým návodem pro programování. Pro složitější nelineární soustavy (statické i dynamické) s více vstupy lze pro popis funkce a pro realizaci využít aparát fuzzy logiky. Modely jsou obvykle realizovány jako systémy Mamdaniho typu nebo Sugeno a Tagakiho typu.

3.2.1. Prediktivní řízení založené na Modelu (MPC)

Na predikci pomocí modelů je založeno řízení Model Predictive Control MPC, jedná se o metodu řízení, jejíž počátky spadají do šedesátých let 20. století a která je mnohokrát zpracovávána v různé literatuře např. [32], [1]. MPC je metoda řízení založená na dynamickém modelu řízené soustavy, a jako je tomu i u jiných pokročilých systému, je její uplatnění v MPC regulátorech, které pracují lépe než klasické PID regulátory, ale opět je zde jasné zaměření pouze na regulaci. MPC regulátor je mnohazměřový regulátor pracující s větším počtem veličin a to jak akčních tak i regulovaných. Reguluje na základě znalostí celé regulované soustavy, u které optimalizuje řídicí vstupy. MPC regulátor není regulátor v klasickém slova smyslu, ale jedná se o algoritmus řízení, který pro svou činnost potřebuje dostatečnou výpočtovou kapacitu, kterou mu zajišťují řídicí systému ať již DCS systémy PLC nebo PC. MPC systém řídí regulovanou soustavu komplexně se znalostí všech vstupů a výstupů řízené soustavy. Přestože MPC řízení je využíváno pro regulaci má své místo i v této práci, protože způsob, jakým pohlíží tento prediktivního regulátoru s modelem na řízenou soustavu, je podobný jako v případě požadavku na predikci celého technologického procesu. Základem je model soustavy a zohlednění vstupů a výstupů této soustavy.

Funkce MPC je založena na vzorkování regulovaných veličin. Při každém vzorku v čase t musí být, mimo příslušné hodnoty vzorku, znám i dynamický model řízené soustavy,



Obr. 6 Základní princip MPC regulátoru

minulé hodnoty regulovaných veličin a minulé hodnoty akčních veličin. Dále musí být k dispozici průběhy požadovaných hodnot regulovaných veličin a to buď skutečné, nebo předpokládané v rámci uvažovaného horizontu predikce¹. Ve své podstatě MPC regulátor nereaguje na základě současné a minulé hodnoty, ale je o několik kroků napřed, tedy reaguje na budoucí hodnotu žádané veličiny a predikovanou hodnotu skutečné veličiny. Princip MPC regulátoru je znázorněn na *Obr. 6*.

Základní princip MPC regulátoru spočívá v tom, že pro regulovaný systém nalezneme odpovídající model, který je podkladem pro optimalizaci hledající neoptimálnější řízení. Tedy obecně řečeno, mějme model soustavy P , který má vstupy u , stavy x a výstupy y . Tento model soustavy je proměnný v čase t .

$$y = P(u, x, t) \quad (3.2)$$

Hledáme optimální řízení, tedy takové vstupy $u_{optimal}$, které minimalizují funkci J

$$u_{optimal} = \arg \min_u J(P(u, x, t), u, t) \quad (3.3)$$

Metoda MPC je detailněji popsána literatuře např. [32].

3.2.2. Postupy při modelování a identifikaci

Základní metody pro vytvoření modelu technologického procesu, ale i modelu obecně jsou dva, analytický a empirický. *Analytický postup*, kdy je model vytvořen čistě matematicky a kdy se jedná o model vnitřní struktury modelovaného procesu, zohledňující všechny (nezanedbatelné) vnitřní a stavové veličiny včetně jejich vzájemných vazeb. Z toho je patrné, že pro takovýto model musíme znát detailně vnitřní strukturu procesu, to ale v některých případech není možné, protože nemusí být známy všechny probíhající procesy a všechny proměnné, stejně tak jsme ve výsledku nuceni přistoupit k určitým zjednodušením a některé parametry zjednodušit, nebo zcela zanedbat abychom byli schopni model realizovat. K určitému zlepšení dojde, pokud modelovaný proces již fyzicky existuje, model potom můžeme stanovit empiricky. Při *empirickém postupu* se nezajímáme o vnitřní strukturu modelované soustavy, ta je pro nás v takovém případě černou skříňkou, která má známé vstupy a výstupy. Měřením, zpracováním a vyhodnocováním vstupních a výstupních veličin

¹ Horizont predikce je relativně krátký časový úsek (čas $t+T$) do kterého algoritmus prediktivního regulátoru provádí výpočet budoucích hodnot regulované soustavy.

soustavy získáme experimentálně model této soustavy, aniž bychom se zajímali o její vnitřní a stavové veličiny. [33]

Aby bylo možné vytvořit model nějakého reálného objektu, je nezbytné přistoupit ke zjednodušení. Model vzniklý na základě těchto zjednodušení zařadíme do třídy dynamických systémů. Rozdělení do tříd dle [33] je následující:

Lineární systémy mají lineární vazby mezi veličinami. **Nelineární systémy** mají alespoň část vazeb nelineárních.

Systémy se soustředěnými parametry mají veličiny nezávislé na poloze v prostoru a jsou proměnné pouze v čase. **Systémy s rozloženými parametry** mají alespoň část veličin proměnných nejen v čase, ale i v prostoru, tzn., že jsou funkcemi času a minimálně jedné prostorové proměnné.

Spojité systémy mají veličiny definovány na spojitém časovém intervalu a jsou spojitými časovými funkcemi. **Diskrétní systémy** mají alespoň část veličin definovaných na množině diskrétních hodnot času a jsou funkcemi v čase diskrétními.

Deterministické systémy jsou systémy, jejichž momentální stav je určen jednoznačně na základě předchozího stavu a působícího vstupu. **Stochastické systémy** mají momentální stav určen pouze s určitou pravděpodobností, do této třídy patří většinou systémy, u kterých na vstupu působí náhodné veličiny.

Jednorozměrové systémy mají jediný vstup a jediný výstup. **Mnohorozměrové systémy** mají více než jeden vstup a minimálně jeden výstup.

Systémy časově variantní mají parametry proměnné v čase, a tedy jejich vlastnosti se v čase mění. **Systémy časově invariantní** jsou systémy, jejichž vlastnosti se v čase nemění.

3.3. Diagnostické hybridní systémy

3.3.1. Konečný automat a technologický proces

Oddělení regulační části od sekvenční není vždy, z pohledu řízení technologického procesu, možné ani žádoucí, proto je potřeba, aby se systém řízení technologického procesu zabýval souběžně jak spojitými, tak i diskrétními procesy. Systém řízení musí být tedy tzv. „Hybridní systém“.

Na technologický proces lze pohlížet jako na soubor jednotlivostí. Jedná se o množinu komponent, které jsou svou povahou diskrétní (pohony, klapky, tlačítka, binární měření...) nebo spojitě (analogová měření, regulace,...). Technologickým procesem se tento soubor stává v okamžiku, kdy jsou jednotlivé komponenty uváděny v činnost ve správném pořadí. Opět lze tedy konstatovat, že technologický proces je hybridním systémem.

Hybridními systémy se zabývají například právě konečné automaty. Teorie konečných automatů jako taková se nezabývá řízením technologického procesu, neboť vznik teorie konečných automatů byl motivován překladači jazyku – rozpoznávání slov. Přesto však lze konečný automat velmi dobře využít jak pro řízení technologického procesu, tak i pro vytvoření jeho modelu. Technologický proces se totiž sestává z měření, kroků a stavů, pro které se jeví vytvoření modelu pomocí konečného automatu jako velmi výhodné. Model vytvořený pomocí konečného automatu je také velmi dobře využitelný pro diagnostiku.

3.3.2. Teorie konečných automatů

Teorií konečných automatů se zabývá velké množství publikací, prostá definice je uvedena např. i v otevřené encyklopedii Wikipedie [34], vlastní teorie je podrobně zpracována v literatuře [35], [36] a [37], další přehledné zpracování je pak v publikacích M.V. Lawsona [38], [39] a [40]. Další zahraniční literatura zabývající se konečnými automaty je [41], [42], [43], [44]. Literatura dostupná v českém jazyce je např. [45], [46], [47], [48], [49], [50]. Značení a názvy z teorie konečných automatů, které ve své práci dále uvádím, jsem použil hlavně z již dostupných českých překladů. Níže uvádím základní údaje a shrnutí potřebná pro využití konečného automatu pro diagnostiku technologického procesu.

Teorie konečných automatů a formálních jazyků je chápána jako součást teorie programování, překladačů, programovacích jazyků a jako součást teoretických modelů reálných elektronických či elektrických, ale i jiných zařízení. Umožňuje návrh, specifikaci a implementaci komunikačních protokolů, reprezentaci dat, formální popis algoritmů, výpočetních procesů, návrh sekvenčních obvodů počítačových systémů, vyhledávání podřetězců v genetické informaci i jinde, obecně studium živých organismů, přirozených jazyků atd. Je disciplínou teoretické (matematické) informatiky.

Níže uvádím pojmy, které se nejčastěji používají v teorii konečných automatů. Definice těchto pojmů je převzata z výše citované literatury.

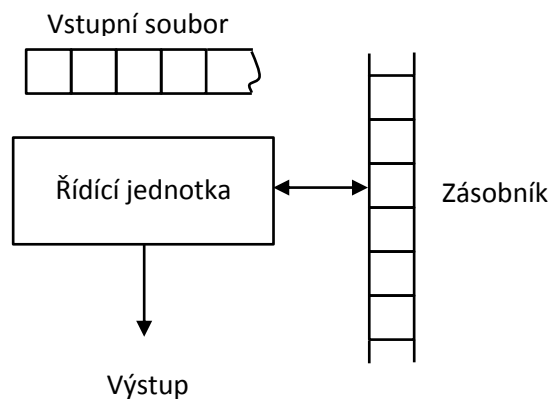
Automat je abstraktní model digitálního počítače, jeho funkce podle [36] je znázorněna na *Obr. 7*. Automat má mechanismus, pomocí kterého čte vstup. Vstupem je řetězec znaků nad abecedou (slovo nad abecedou). Tento vstup, který je na obrázku označen jako „Vstupní soubor“, může být automatem přečten, ale nemůže být měněn. Vstupní soubor je rozdělen na buňky, kdy každá může obsahovat pouze jeden symbol. Automat je vybaven dočasným „Zásobníkem“, který má nekonečný počet buněk, jejichž obsah může být automatem měněn. Automat má dále „Řídící jednotku“, která určuje, ve kterém z konečné množiny interních stavů se automat nachází. Stav může automat měnit pouze definovaným způsobem.

Abeceda je konečná neprázdná množina symbolů (písmen), označuje se Σ

Slovo nad abecedou Σ je konečná posloupnost symbolů z abecedy Σ . Prázdnou posloupnost, tedy prázdné slovo označujeme ε

Délka slova je délka posloupnosti symbolů (písmen) které slovo vytváří. Délku slova označujeme $|w|$.

Jazyk nad abecedou Σ je libovolná množina slov nad abecedou Σ .



Obr. 7 Automat

Konečný automat (označovaný zkratkou **KA**, také **FSM** z anglického *finite state machine*, případně **DFA** z anglického *deterministic finite automation*) je abstraktní (virtuální) systém s konečným počtem stavů (odtud název konečný automat), používaný v informatice pro studium formálních jazyků. Popisuje velice jednoduchý počítač, který může být v jednom z několika definovaných stavů, mezi kterými přechází na základě symbolů, které čte ze vstupu. Konečný automat nemá žádnou další paměť kromě informace o aktuálním stavu. Konečný automat je velice jednoduchý výpočetní model, dokáže rozpoznávat pouze regulární jazyky¹. Konečné automaty se používají pro zpracování regulárních výrazů, např. jako součást lexikálního analyzátoru v překladačích. V informatice rozlišujeme automat Mealyho a Mooreův.

Činnost konečného automatu můžeme zjednodušeně popsat jako automat nacházející se v definovaném počátečním stavu. V každém kroku konečného automatu se přečte vstupní symbol a na základě tohoto symbolu přejde automat do nového stavu definovaného přechodovou funkcí. Ve čtení symbolů automat pokračuje, dokud se nedostane do konečného stavu. Opět při tomto popisu KA je zřejmé, že takovýto popis vyhovuje popisu technologického procesu.

¹ Kleenova věta: Libovolný jazyk je regulární, právě když je rozpoznatelný konečným automatem.

Definice konečného Automatu - Konečné automaty můžeme dělit podle různých hledisek, pro účely definice se zaměřím na rozdělení na konečné automaty deterministické (DFA) a nedeterministické (NFA). Přejed z jednoho stavu do následujícího je určen stavem, ve kterém se automat nachází a symbolem, který přichází na vstup (nebo který je čten na vstupu).

Deterministický konečný automat je automat, který má jednoznačně definovaný přechod z aktuálního stavu do nového po příchodu vstupního symbolu. Přejed je určen stavem, ve kterém se automat nachází a symbolem, který přichází na vstup (nebo který je čten na vstupu). Pro každý přechod existuje právě jeden cílový stav.

Deterministický konečný automat podle definice v literatuře [36], [37] a se zohledněním značení v české literatuře zejména [45], [49], je každá pětice prvků

$$A = (Q, \Sigma, \delta, q_0, F) \quad (3.4)$$

kde:

Q je konečná neprázdná množina stavů (interních stavů)

Σ je konečná neprázdná množina vstupních symbolů (vstupní abeceda).

δ je přechodová funkce, $\delta: Q \times \Sigma \rightarrow Q$.

q_0 je počáteční (iniciální) stav, $q_0 \in Q$.

F je množina koncových, neboli přijímajících stavů, $F \subseteq Q$.

Vstupní (čtecí) mechanismus se pohybuje pouze směrem zleva doprava a v každém kroku může přečíst pouze jeden symbol. Přejed do nového stavu je řízen přechodovou funkcí δ . Zápis $\delta(q_0, a) = q_1$ znamená, že pokud se DFA nachází ve stavu q_0 a vstupní symbol, který je právě čten je a potom DFA přejde do nového stavu q_1 .

Nedeterministický konečný automat nemusí mít jasně definovaný nový stav, do kterého se nastaví po přečtení vstupního symbolu, ale je pouze vymezena množina stavů do kterých může přejít. Pro každý přechod existuje celá množina cílových stavů.

Nedeterministický konečný automat podle definic v [36], [37],[45],[49], je každá pětice prvků

$$A = (Q, \Sigma, \delta, q_0, F) \quad (3.5)$$

kde:

Q je konečná neprázdná množina stavů.

Σ je konečná neprázdná množina vstupních symbolů (vstupní abeceda).

δ je přechodová funkce, $\delta: Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow 2^Q$
 ε je prázdný symbol a 2^Q je potenční množina.

q_0 je počáteční (iniciální) stav, $q_0 \in Q$.

F je množina koncových, neboli přijímajících stavů, $F \subseteq Q$

Hlavní rozdíl mezi DFA a NFA je ten, že $\delta(q_0, a) = \{p_1, p_2, \dots, p_n\}$, tedy pokud se NFA nachází ve stavu q_0 a vstupní symbol, který je právě čten je a , potom NFA se může rozhodnout do kterého stavu, z dané množiny, se dostane.

Při realizaci konečného automatu se nejčastěji používají odvozené automaty. Jedná se o konečné automaty s výstupem, typu Moore a Mealy.

Sekvenční obvod typu Moore. Jeho základem je podle literatury [36], [45] konečný automat s výstupem, jedná se o jednoduché zařízení, které má konečný počet vnitřních stavů, mezi nimiž automat přechází na základě symbolů vstupní abecedy. Každý vnitřní stav má definovanou jednu hodnotu na výstupu. Výchozí vnitřní stav automatu, ve kterém se automat nachází před zadáním prvního vstupního symbolu a pravidla pro přechody mezi jednotlivými stavy, musí být definovány. Změna na vstupu se projeví na výstupu v následujícím stavu. Výstupní funkce jsou funkcemi vnitřního stavu ($\lambda: Q \rightarrow Y$)

Konečný automat typu Moore (definice dle [36]¹) je uspořádaná šestice

$$(Q, \Sigma, O, \delta, \lambda, q_0) \quad (3.6)$$

kde:

Q je konečná neprázdná množina vnitřních stavů

Σ je konečná neprázdná množina vstupních symbolů (vstupní abeceda)

O je konečná neprázdná množina výstupních symbolů (výstupní abeceda)

δ je přechodová funkce $\delta: Q \times \Sigma \rightarrow Q$

λ je výstupní funkce $\lambda: Q \rightarrow O$

q_0 je výchozí vnitřní stav, pro který platí $q_0 \in Q$

Pokud je výstupní funkce $F(t)$ závislá pouze na aktuálním stavu $q(t)$ a je nezávislá na aktuálním vstupu, potom je výstupní funkce dána jako

$$F(t) = \lambda(q(t)) \quad (3.7)$$

Sekvenční obvod typu Mealy podle [36], jeho základem je konečný automat s výstupem a zobecnění sekvenčního obvodu typu Moore, od kterého se liší tím, že výstup závisí nejen na vnitřním stavu, ale i na vstupu. Tedy do výstupní funkce vstupuje i příslušný prvek vstupní abecedy jako parametr ($\lambda: Q \times \Sigma \rightarrow O$).

Konečný automat typu Mealy (definice dle [36]) je uspořádaná šestice

$$(Q, \Sigma, O, \delta, \lambda, q_0) \quad (3.8)$$

¹ V odborné literatuře je používáno různé značení jednotlivých množin, stejně tak Mooreův automat je někdy definován jako uspořádaná šestice, někdy jako uspořádaná sedmice (včetně množiny koncových stavů). Pro zachování obecných principů jsem použil citovanou literaturu jako podklad pro zde uváděné definice.

kde:

- Q je konečná množina vnitřních stavů
- Σ je konečná množina vstupních symbolů (vstupní abeceda)
- O je konečná neprázdná množina výstupních symbolů (výstupní abeceda)
- δ je přechodová funkce $\delta: Q \times \Sigma \rightarrow Q$
- λ je výstupní funkce $\lambda: Q \times \Sigma \rightarrow O$
- q_0 je výchozí vnitřní stav, pro který platí $q_0 \in Q$

Pokud je výstupní funkce $F(t)$ závislá na aktuálním stavu $q(t)$ a zároveň na aktuálním vstupu $x(t)$, potom je výstupní funkce dána jako

$$F(t) = \lambda(q(t), x(t)) \quad (3.9)$$

3.3.3. Konečný automat pro diagnostiku

Na základě teorie KA uvedené v předchozím odstavci nyní představím definici konečného automatu využitelného pro řízení technologického procesu, a který budu dále v textu označovat jako „Technologický konečný automat“. Tento automat představuje jednoduchý automat, který je obrazem technologického procesu. Technologický proces nenabývá neznámých hodnot, neboť je vždy navržen pro určitou technologii a na základě známých, respektive požadovaných, technologických postupů. V případě, že technologie pracuje podle svého účelu užití, musí se za stejných podmínek také stejně chovat. Technologický konečný automat může nabývat definovanou-konečnou množinu hodnot (stavů). Po přečtení vstupních dat tj. po přečtení vstupního symbolu (Σ), přejde technologický konečný automat do nového stavu. Technologický proces, nebo jednotlivé podprocesy, které modelujeme, jsou prováděny postupně krok po kroku (stav \rightarrow přechod za předpokladu splněných podmínek \rightarrow nový stav \rightarrow přechod za předpokladu, ... \rightarrow atd...) proto na technologický konečný automat můžeme pohlížet jako na deterministický konečný automat. Pro účely diagnostiky je technologický konečný automat doplněn o množinu minimálních a maximálních časů. Potom tedy definuji Technologický konečný automat takto:

Technologický konečný automat je každá sedmice prvků

$$A = (Q, \Sigma, T_{min}, T_{max}, \delta, q_0, F) \quad (3.10)$$

kde:

Q je konečná neprázdná množina stavů.

Σ je konečná neprázdná množina vstupních symbolů (vstupní abeceda).

T_{min} množina minimálních časů

T_{max} množina maximálních časů

δ je přechodová funkce, $\delta: Q \times \Sigma \rightarrow Q$.

q_0 je počáteční (iniciální) stav, $q_0 \in Q$.

F je množina koncových, neboli přijímajících stavů, $F \subseteq Q$

K přechodu do nového stavu dojde, jsou-li splněny podmínky pro přechod (je splněna přechodová funkce) a zároveň čas pro předchozí krok dosáhl minimálně času T_{min} . Pokud je čas menší než T_{min} technologický proces počká, až bude splněn čas, jedná se vlastně o další podmínku přechodu. Čas T_{min} je jednou z podmínek přechodu (je potřeba pro řídicí technologický konečný automat), další definovaný čas T_{max} je indikací chyby (indikace pro diagnostický konečný automat). Množina koncových stavů F je v případě technologického konečného automatu pouze indikací stavu ukončení technologického procesu.

V předchozím odstavci se zmiňuji o diagnostickém konečném automatu, tím myslím konečný automat popisující technologický proces a jehož výstupem jsou údaje pro diagnostiku. Tento technologický konečný automat pro diagnostiku (v textu dále uváděn jako diagnostický konečný automat) je navržen dalším rozšířením technologického konečného automatu. Jak bylo uvedeno dříve, je množina maximálních časů T_{max} , která je v definici technologického automatu uvedena pouze jako podmínka vyjmutá z přechodové funkce δ , potřeba pro účely diagnostiky. Pro diagnostiku zavádím mimo množiny časů T_{max} ještě další množinu a to množinu chybových stavů kterou označím „ E “. Jedná se o konečnou množinu stavů, ve kterém zakončí svou činnost technologický proces při nestandardním (poruchovém)

stavu. Z chybového stavu E lze odvodit důvody nestandardního chování/poruchy. Případně lze stav E využít pouze pro indikaci výskytu poruchy a důvody jejího vzniku odvodit podle stavu, ze kterého systém do stavu E přešel.

Diagnostický konečný automat potom definuji jako každou osmici prvků

$$A = (Q, \Sigma, T_{min}, T_{max}, \delta, q_0, F, E) \quad (3.11)$$

kde:

Q je konečná neprázdná množina stavů.

Σ je konečná neprázdná množina vstupních symbolů (vstupní abeceda).

T_{min} množina minimálních časů

T_{max} množina maximálních časů

δ je přechodová funkce, $\delta: Q \times \Sigma \rightarrow Q$.

q_0 je počáteční (iniciální) stav, $q_0 \in Q$.

F je množina koncových, neboli přijímajících stavů, $F \subseteq Q$

E je množina stavů při poruše/chybě, $E \subseteq Q$

V případě, že diagnostický konečný automat doplníme o množinu výstupů, stává se z konečného automatu řídicí konečný automat, který může svými výstupy ovládat proces (sepnutí ochranných funkcí, změna parametrů regulace a pod) tato problematika je však již nad rámec této práce.

Hlavním tématem mé práce je prediktivní diagnostika technologických procesů. Pro tuto diagnostiku využiji výše definovaný diagnostický konečný automat dle (3.11), který ještě dále, pro účely diagnostiky, rozšířím o novou množinu, kterou označím α . Množina α je tvořena posloupností údajů (časy, naměřené hodnoty,...) a je definovaná pro každý predikovaný stav. Jedná se o časovou řadu hodnot, které se dále vyhodnocují vhodnou prediktivní metodou, (např. lineární regresi). Vyhodnocení těchto hodnot znamená určení

trendu. Ze získaného trendu lze potom predikovat události, které mohou v technologickém procesu nastat.

3.3.4. Konečný automat pro prediktivní diagnostiku

Konečný automat pro prediktivní diagnostiku vychází z definic konečného automatu, uvedených v odstavci 3.3.2, a rozšířením běžných konečných automatů dle odstavce 3.3.3. Zde uvádím definici Konečného automatu pro prediktivní diagnostiku s podrobným vysvětlením, co jednotlivé množiny představují v rámci technologického procesu.

Mějme diagnostický konečný automat definovaný osmicí prvků

$$A_{pd} = (Q, \Sigma, T_{min}, T_{max}, \delta, q_0, F, E) \quad (3.12)$$

a množinou časových řad α ,

kde:

Q představuje všechny stavy technologického procesu a to jak standardní (do kterých přejde technologický proces pracující správně, pro účel pro který byl navržen) tak i poruchové (do kterých přejde v případě nestandardního chování technologického procesu), Např. zařízení je připraveno, probíhá plnění nádrže, maximální teplota zařízení byla překročena,...

Σ představuje události¹, které mohou nastat v technologickém procesu a které způsobí přechod mezi stavy. Z pohledu technologického procesu to jsou vstupy do tohoto procesu, jejich vzájemné kombinace a dále např. počítané vnitřní proměnné, které způsobí přechod. Např. povel start a zároveň zavření hlavní ventil; čerpadlo vypnuto a zároveň zavření ventil a zároveň hladina menší než daná mez;... Množinu všech jednotlivých vstupů a vnitřní proměnných (tedy bez vzájemných kombinací) označím jako Σ^Σ Např. zmáčknu tlačítko start, dosažena limitní hladina, vypnuto čerpadlo,...Je-li:

$$\Sigma^\Sigma = \{i_1, i_2, i_3, i_4\}$$

¹Technologický proces a stejně tak i jeho model, jsou řízeny událostmi, pro model realizovaný konečným automatem tedy množina událostí odpovídá množině vstupních symbolů.

$$a = i_1 \cap i_2$$

$$b = i_1 \cap i_2 \cap i_3$$

$$c = i_3 \cap i_4$$

potom:

$$\Sigma = \{a, b, c\}$$

- T_{min} jak bylo uvedeno již v předchozím odstavci, jedná se vlastně o jednu z podmínek pro přechod do následujícího stavu a to ať se jedná o minimální čas přechodu, nebo minimální čas trvání stavu. Minimální doba trvání výchozího stavu q_0 a koncového stavu q_n nemá většinou žádný technický význam.
- T_{max} jedná se o indikaci chybového stavu, o tom, že přechod z jednoho stavu do následujícího, nebo i vlastní definovaný stav, trval déle, než při správné funkci všech komponent technologického procesu měl. Maximální doba trvání výchozího stavu q_0 a koncového stavu q_n opět nemá žádný podstatný technický význam.
- δ ($\delta: Q \times \Sigma \rightarrow Q$) představuje přechodovou funkci, kdy technologický proces přechází z jednoho stavu do dalšího. Z pohledu technologického procesu se po dobu přechodu přestavují zařízení do pozice požadované novým stavem q_i , (zavírají se ventily, zapínají čerpadla,...)
- q_0 ($q_0 \in Q$) je výchozí (iniciální) stav technologického procesu. Informuje nás, že technologický proces je připraven a čeká na impuls k zahájení činnosti. Např. technologické zařízení je připraveno a čeká na povel start.
- F ($F \subseteq Q$) je koncový stav technologického procesu při jeho správné funkci. Např. byl vyroben požadovaný produkt v žádaném množství a kvalitě.
- E ($E \subseteq Q$) jsou stavy technologického procesu, do kterých přejde při nesprávné funkci. Např. překročena limitní mez, překročen maximální čas operace,...

α zpočátku prázdná množina, definovaná pro každý diagnostikovaný uzel¹, která se v průběhu procesu plní údaji pro potřeby prediktivní diagnostiky. Např. doba trvání daného přechodu, nebo stavu, případně další diagnostické hodnoty dosažené v průběhu technologického procesu.

Množina α pro predikci bude tvořena zejména z časových údajů (doba přechodu a doba trvání stavu) a dále z analogových měření, která mají přímý vztah k technologickému procesu a jejichž sledování má význam pro diagnostiku. Měření lze v technologickém procesu diagnostikovat z pohledu správné funkce senzoru, to je možné např. diagnostikou pro údržbu, kdy zjišťujeme jak je dané měření spolehlivé, zda údaje z měření jsou spolehlivé, případně čidlo neměří správně a požaduje údržbu apod. (tzv. kredibilita/diskredibilita senzoru). Diagnostika samotných měření a čidel není předmětem této práce, pro naše účely předpokládáme, že čidla předávají správné údaje, tedy kredibilita senzoru je v pořádku. Analogová měření, která je vhodné pro účely prediktivní diagnostiky sledovat jsou měření přímo vztažená k technologickému procesu a jejichž údaje mají charakter vypovídající o stavu technologického procesu.

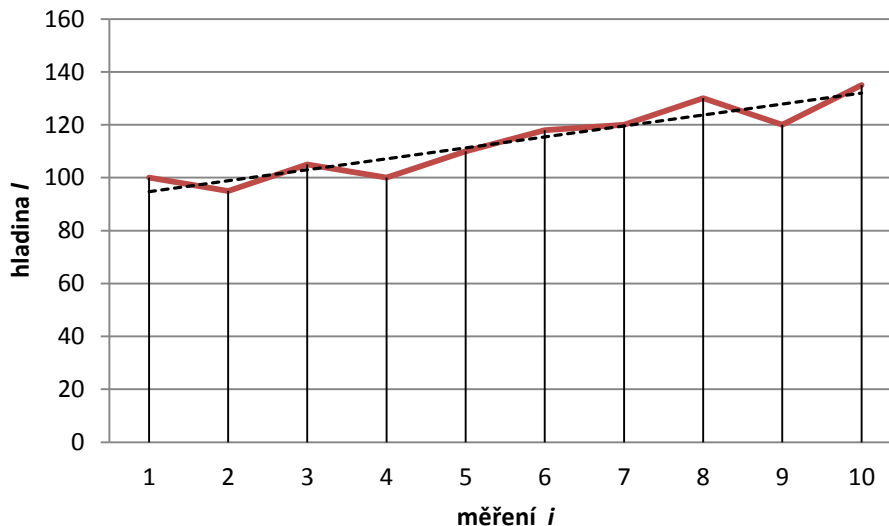
Příklad takového analogového měření vztaženého k technologickému procesu, které má význam pro diagnostiku:

Mějme nádrž, kterou na základě požadavků technologického procesu, budeme plnit po pevně stanovený čas t a současně v této nádrži měříme hladinu l . Pokaždé kdy konečný automat přejde do stavu q_i , technologického procesu, který znamená dosažení času t při plnění nádrže, uložíme hodnotu hladiny v nádrži l_{qi} do paměti. Takto budeme plnit řadu měření α , příslušející technologicky k naplnění nádrže.

Např. řada:

$$\alpha_l = \{100; 95; 105; 100; 110; 118; 120; 130; 120; 135\}$$

¹Diagnostikovaný uzel je takové místo v modelu technologického procesu, které podrobujeme diagnostice a ze kterého sbíráme diagnostická data.



Obr. 8 Zpracování dat z množiny α , měření v čase α_{T_i} a linearizovaný trend

Předpokládáme-li, že na výstup ze snímače hladiny se můžeme spolehnout (diagnostika čidla, zdvojené měření, výběr 2 ze 3 apod.), potom údaje o výšce hladiny po uplynutí času plnění $\{\alpha_i\}$ jsou právě údaje přímo vztažené k technologickému procesu. Tyto údaje jsou vyneseny do grafu na Obr. 8 a to včetně jejich linearizovaného trendu. Z grafu je patrné, že přestože při každém kroku nádrž plníme po stejný časový úsek, postupně dochází ke zvyšování hladiny a v budoucnu by mohlo dojít k situaci, kdy bude nádrž přeplněna. Za předpokladu, jak bylo uvedeno výše, že všechna měření jsou správná, je tento nárůst hladiny způsoben technologickým procesem a je nutné provést jeho kontrolu. Je možné, že nedochází k úplnému vypuštění nádrže před novým napouštěním, případně došlo ke změně průtoku dávkovaného materiálu apod.

3.3.4.1. Příklad konečného automatu pro prediktivní diagnostiku

Mějme technologický proces s definovanými stavy q_0, q_1, q_2, q_3 , přičemž stav q_0 je počáteční stav, ze kterého technologický proces vychází a q_3 je konečný stav ve kterém technologický proces ukončí řádně svou činnost, $\{E\}$ kde $E \subseteq Q$, představuje konečnou množinu stavů (může to být i pouze jediný stav q_E , kde $q_E \in E$) do kterých přejde technologický proces v případě nestandardního chování procesu (poruchy). Vstupy technologického procesu $\Sigma^{\mathcal{I}}$ (množina vstupů) jsou $i_1, i_2, i_3, i_4, i_5, i_6$ (vstupní měření, nebo

vnitřní počítané hodnoty). Vstupními symboly, tedy událostmi v technologickém procesu jsou události způsobené vstupy, např.:

$$\begin{aligned}
 a &= i_1 & a_E &= i_1 \cap i_2 \cap i_3 \\
 b &= \bar{i}_1 \cap i_2 \cap i_3 & b_E &= i_1 \\
 c &= \bar{i}_2 \cap i_3 \cap i_4 & c_E &= i_1 \cap \bar{i}_2 \cap \bar{i}_3 \\
 & & d_E &= \bar{i}_6
 \end{aligned} \tag{3.13}$$

Vstupní abeceda Σ je potom $(a, b, c, a_E, b_E, c_E, d_E)$. T_{min} jsou minimální časy přechodu technologického procesu pro přechod do nového stavu a minimální časy trvání stavu až do okamžiku přechodu do nového stavu. V technologicky odůvodněných případech je $T_{min} = 0$. Jsou to případy, kdy technologický proces není nijak limitován minimálními časy. T_{max} jsou maximální časy pro přechod z jednoho stavu do následujícího a maximální časy pro setrvání technologického procesu v daném stavu. V technologicky odůvodněných případech je $T_{max} = +\infty$. Jsou to případy, kdy technologický proces není nijak limitován maximálními časy přechodu z jednoho stavu do druhého, nebo trváním stavu. Přechodová funkce δ představuje splnění podmínek pro přechod technologického procesu do nového stavu. Na základě přečtení vstupních symbolů, tedy událostí v technologickém procesu, technologický proces přejde do nového stavu, pokud jsou definované podmínky pro přechod splněny a zároveň, jak je uvedeno výše, je pro tento přechod splněna podmínka dodržení minimálního času T_{min} (pokud existuje). Jestli-že nejsou splněny podmínky pro přechod, anebo pokud jsou splněny, ale nebyl dosažen minimální čas T_{min} , proces setrvá v posledním dosaženém stavu. Proces přejde do definovaného poruchového stavu, pokud je překročen maximální čas, anebo pokud technologický proces v diagnostikovaném uzlu hlásí nestandardní/poruchové chování, viz (a_E, b_E, c_E, d_E) ve (3.13).

Výše slovně popsaný vzorový technologický proces mohu namodelovat pomocí konečného automatu pro prediktivní diagnostiku, standardními způsoby popisu konečného automatu, jak je uvedeno v různé literatuře např. [45], [46], [49], avšak je třeba zohlednit odlišnosti mezi běžným konečným automatem a konečným automatem pro prediktivní diagnostiku.

Popis může být uveden následujícími způsoby:

- výčtem prvků, jako osmicí a časovou řadu,
- tabulkou a časovou řadou,
- stavovým diagramem,
- stavovým stromem.

Výčet prvků a časová řada:

$$A_{pd} = (Q, \Sigma, T_{min}, T_{max}, \delta, q_0, F, E) \quad (3.14)$$

$$\alpha_i = \{\alpha_{Ti}\} \quad (3.15)$$

tedy podle uvedeného zadání

$$A_{pd} = ((q_0, q_1, q_2, q_3), (a, b, c, a_E, b_E, c_E, d_E), T_{min}, T_{max}, \delta, q_0, q_3, \{q_E\}) \quad (3.16)$$

$$\alpha_i = \{\alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn}\} \quad (3.17)$$

Index i ve vztahu (3.15) resp. (3.17) představuje i -tý přechod a i -tý stavy.

Tabulka:

Příslušný technologický proces, je popsán tabulkou uvedenou na *Obr. 9* a časovou řadou (3.18) pod tabulkou. Znaky v tabulce představují stav události. **1** znamená, že událost byla splněna (např. ventil v poloze otevřeno), **0** znamená, že událost není splněna (opačný stav, např. ventil v poloze zavřeno), **x** ve sloupci dané události znamená, že stav této události v příslušném stavu q_i do technologického procesu nijak nezasahuje a na její hodnotě z pohledu funkčnosti modelu daného technologického procesu nezáleží. Prázdná políčka pak znázorňují, že příslušná data nejsou relevantní, nebo neexistují. Ve sloupci Σ uvádím odpovídající symboly vstupní abecedy.

$Q \backslash \Sigma^2$	i_1	i_2	i_3	i_4	i_5	i_6	Σ	<i>přechod</i>
q_0	1	x	x	x	x	x	a	$\delta_1: q_0 \rightarrow q_1$
	1	1	1	x	x	x	a_E	$\delta_{1E}: q_0 \rightarrow q_E$
q_1	0	1	1	x	x	x	b	$\delta_2: q_1 \rightarrow q_2$
	1	x	x	x	x	x	b_E	$\delta_{2E}: q_1 \rightarrow q_E$
q_2	x	1	1	1	x	x	c	$\delta_3: q_2 \rightarrow q_3$
	x	1	0	0	x	x	c_E	$\delta_{3E}: q_2 \rightarrow q_E$
q_3								
	x	x	x	x	x	0	d_E	$\delta_{4E}: q_3 \rightarrow q_E$

Obr. 9 Konečný automat pro prediktivní diagnostiku – tabulka

Dále definuji diagnostikované uzly a pro ně příslušné časové řady α_i . Diagnostikované uzly tedy místa, která, jsou pro potřeby diagnostiky zajímavá, jsou doby trvání stavů q_1, q_2 , které označím jako $\alpha_{q_1}, \alpha_{q_2}$ a dále pak doba trvání přechodu mezi stavy $q_0-q_1, q_1-q_2, q_2-q_3$, které označím jako $\alpha_{q_0-q_1}, \alpha_{q_1-q_2}, \alpha_{q_2-q_3}$.

Potom časové řady pro tyto diagnostikované jsou:

$$\alpha_{q_0-q_1} = \delta_1 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

$$\alpha_{q_1} \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

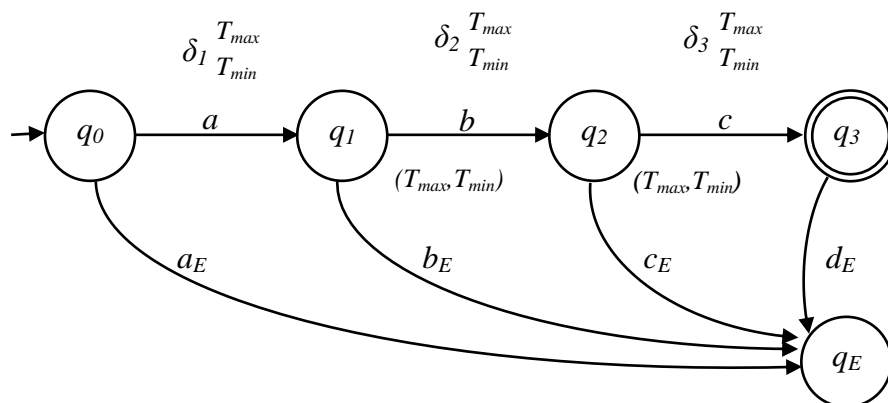
$$\alpha_{q_1-q_2} = \delta_2 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \} \tag{3.18}$$

$$\alpha_{q_2} \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

$$\alpha_{q_2-q_3} = \delta_3 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

Stavový diagram:

Popis technologického procesu stavovým diagramem je uveden na *Obr. 10*. Stav q_0 představuje výchozí stav, ve kterém technologický proces začíná svou činností, δ_i představuje přechod do nového stavu q_i , přechod do poruchového stavu q_E , který je v tomto ukázkovém případě technologického procesu jeden společný, způsobí události a_E, b_E, c_E, d_E . Splnění podmínek pro přechod do nového stavu je stejné, jako je uvedeno v tabulce *Obr. 9*. Časové řady α_i jsou stejné jako v případě znázornění tabulkou (3.18).

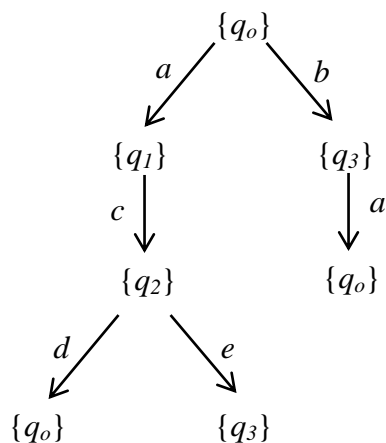


Obr. 10 Konečný automat pro prediktivní diagnostiku – stavový diagram

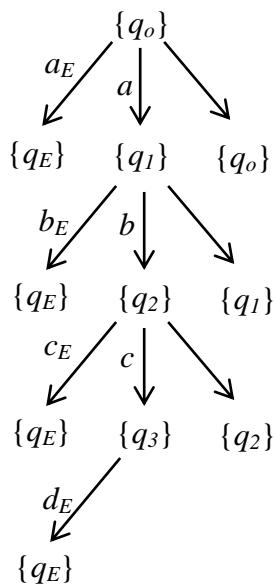
Stavový strom:

Stavový strom je takový popis konečného automatu, který se hodí pro automaty, u nichž je možné každého stavu dosáhnout z počátečního stavu, s tím že o nedosažitelné stavy se vůbec nezajímáme. Vlastně se jedná o stavový diagram, kde každý stav má znázorněn graficky všechny stavy, kterých může dosáhnout. Obecný příklad takového stavového stromu je na *Obr. 11*. Na *Obr. 12* je pak, pouze pro ilustraci, uveden stavový strom pro náš případ popisu technologického procesu. Časové řady α_i jsou stejné jako v případě znázornění tabulkou (3.18). Vzhledem k tomu, že technologický proces, jak jsem již vícekrát zmínil, se

sestává z po sobě jdoucích událostí, nemá popis stavovým stromem pro modelování technologického procesu podstatný význam.



Obr. 11 Konečný automat - Stavový strom



Obr. 12 Konečný automat pro prediktivní diagnostiku - stavový strom

3.4. Konečný automat pro prediktivní diagnostiku jako model technologického procesu

Jak jsem již dříve uvedl, technologický proces je souhrn kroků, podmínek a regulací, které jsou prováděny v daném pořadí, čase a podle požadavků tohoto procesu. Pro takto definovaný technologický proces mohu vytvořit jeho model za použití konečného automatu.

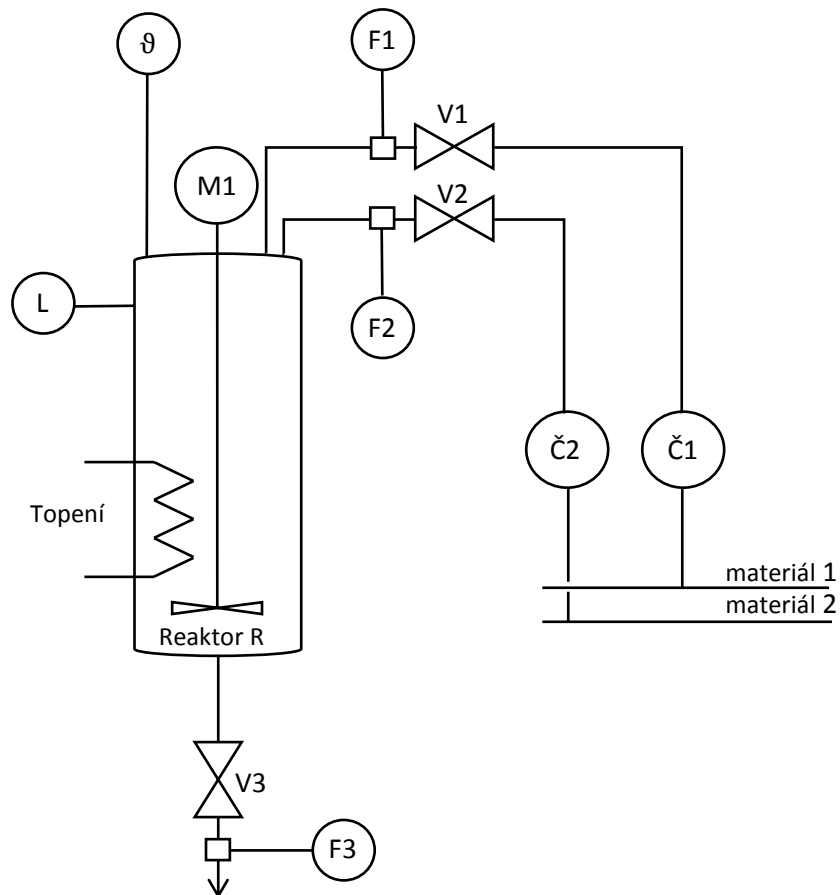
Při tvorbě modelu je nezbytně nutné si uvědomit, že netvořím model zařízení jako např. model tepelného výměníku, nebo model reaktoru apod. ale tvořím model technologického procesu, tedy model procesu ohřívání média, nebo model výroby nějakého produktu (např. polymeru, směsi olejů,...) apod.

Jaký je vztah mezi technologickým procesem a konečným automatem? Jak jsem již dříve uvedl, má technologický proces konečný počet vstupů, výstupů a stavů z tohoto pohledu jej potom mohu ztotožnit s konečným automatem. Dále mohu na základě obecných principů konstatovat, že technologický proces má jednoznačně definovaný přechod z aktuálního stavu do nového po příchodu vstupního symbolu (události), tzn., že technologický proces odpovídá deterministickému konečnému automatu (viz definice KA v kapitole 3.3.2 této práce). Vzhledem k tomu, že výstupem technologického procesu, z pohledu definice konečného automatu, je obecně signalizace o řádném, případně chybném ukončení daného technologického procesu, není třeba pro model technologického procesu konečným automatem řešit automat s výstupem, pokud by bylo u technologického procesu třeba řešit výstup, byl by jeho modelem Mealyho konečný automat, protože výstupní funkce technologického procesu závisí jak na stavu, tak na symbolu vstupní abecedy, pro výstupní funkci δ by potom platilo $\delta: Q \times \Sigma \rightarrow Y$.

3.4.1. Návrh modelu technologického procesu

Technologický proces se vždy skládá z činností, které se provádějí v určitém pořadí v určitém čase a mají definované vazby, které lze pokládat za pevné. Technologický proces je možné rozdělit na jednotlivé subprocesy, které jsou vzájemně provázány a mohou probíhat současně, nebo po sobě. Tyto subprocesy mají vazby na některou část technologického procesu a mají jasně definovaný začátek a konec. Před vlastním vytvořením modelu technologického procesu je potřeba technologický proces rozdělit na tyto dílčí subprocesy,

pro které budou vytvořeny modely využitelné pro predikci. Protože se zabýváme pouze predikcí procesu, nikoliv jeho řízením, není třeba vytvářet model celého technologického procesu, ale jen těch částí (subprocesů), které jsou pro technologický proces a zejména jeho diagnostiku, rozhodující.



Obr. 13 Příklad technologického procesu – výroba produktu A

3.4.1.1. Příklad technologického procesu 1 – Výroba produktu „A“

Na Obr. 13 uvádím příklad takového dílčího technologického procesu. Uvedený proces se zabývá výrobou produktu „A“ v reaktoru R, ze vstupních materiálů 1 a 2. Vstupní materiály jsou dodávány do reaktoru čerpadly Č1 a Č2 přes ventily V1 a V2. Množství vstupního materiálu se měří průtokoměry F1 a F2. V reaktoru R1 je míchadlo ovládané motorem M1 a topení (případně chlazení). V reaktoru je měřena teplota θ a hladina L. Hotový

produkt je vypouštěn z reaktoru ventilem V3. Množství konečného produktu je při vypouštění měřeno průtokoměrem F3.

Slovní popis výroby produktu A.

Po zadání povelu na zahájení výroby Produktu „A“, za podmínky, že reaktor R je prázdný, teplota v reaktoru je menší než ϑ_1 a ventil V3 je zavřený, otevře ventil V1 a spustí se čerpadlo Č1. Po dosažení množství F_a materiálu 1 v reaktoru zavře ventil V1 a vypne čerpadlo Č1, zároveň otevře ventil V2 a sepne čerpadlo Č2, které začne dávkovat materiál 2 do reaktoru, současně sepne míchadlo M1. Po dosažení množství F_b materiálu 2 v reaktoru zavře ventil V2 a vypne čerpadlo Č2. Za stálého chodu míchadla sepne topení, které zvýší teplotu v reaktoru na teplotu ϑ_R . Po dobu T , za chodu míchadla je udržována teplota ϑ v reaktoru na hodnotě $\vartheta_R \pm 5^\circ\text{C}$. Po uplynutí času T otevře ventil V3 a produkt A se z reaktoru vypustí. Snímač množství F3 měří množství produktu „A“. Po vypuštění nádrže, které je odvozeno od signalizace hladiny v nádrži $l < L_0$ vypne topení a po poklesu teploty v reaktoru pod hodnotu ϑ_2 zavře ventil V3. Proces přejde do výchozího stavu.

3.4.1.1.1. Tvorba modelu

Model technologického procesu, popsaného v odstavci 3.4.1.1, vytvořím pomocí konečného automatu pro diagnostiku technologického procesu definovaného v odstavci 3.3.4.

$$A_{pd} = (Q, \Sigma, T_{min}, T_{max}, \delta, q_0, F, E) \quad (3.19)$$

$$\alpha_i = \{ \alpha_{Ti} \} \quad (3.20)$$

Nejdříve určím jednotlivé množiny, technologického procesu, které vstupují do modelu. Vytvářím model procesu, nikoliv zařízení a také je nutné se zmínit, že se nejedná o model vlastního řízení technologického procesu. Předpokládám, že technologie je řízena samostatným řídicím automatem a pro diagnostiku není vůbec důležitá metoda nebo systém řízení vlastní technologie.

Stavy technologického procesu Q:

- Proces je připraven stav q_0
- Napouštění materiálu 1 stav q_1

- Napouštění materiálu 2 stav q_2
- Reakce stav q_3
- Stáčení produktu A stav q_4
- Výroba ukončena stav q_5
- Porucha výroby stav q_E

Neprázdňá konečňá množina stavů je potom $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_E\}$

Vstupy technologického procesu Σ^Σ :

- povel start výroby! vstup i_1
- reaktor R je prázdňý: vstup i_2
- teplota v reaktoru $< \vartheta_1$: vstup i_3
- V3 zavřen: vstup i_4
- Průtok $F_1 \geq F_a$ vstup i_5
- Průtok $F_2 \geq F_b$ vstup i_6
- čas T vstup i_7

Události neboli vstupní abeceda Σ , je potom konečňá množina událostí způsobená vstupy $\Sigma^\Sigma \{i_1, i_2, i_3, i_4, i_5, i_6, i_7\}$ viz dále.

Množina časů T_{min}, T_{max} :

Minimální a maximální časy budou určeny pro jednotlivé přechody a stavy, pro které jsou smysluplné. Tyto časy musí být určeny ve spolupráci s technologií, nebo musí být odzkoušeny na reálném, správně fungujícím zařízení a to při zohlednění všech provozních podmínek. Pro tento vzorový příklad budou časy určeny.

Podmínky pro přechod stavů δ :

Pro vstupy $\Sigma^\Sigma = \{i_1, i_2, i_3, i_4, i_5, i_6, i_7\}$ uvádím níže přechodové funkce pro přechody mezi stavy s tím, že x značí libovolnou hodnotu vstupu i_n , tyto funkce vycházejí ze slovního popisu výroby v odstavci 3.4.1.

$$\delta_1: q_0 \rightarrow q_1: \quad \delta(q_0, 1111xxx) = q_1$$

$$\delta_2: q_1 \rightarrow q_2: \quad \delta(q_1, xxxxlxx) = q_2$$

$$\delta_3: q_2 \rightarrow q_3: \quad \delta(q_2, xxxxxx1x) = q_3$$

$$\delta_4: q_3 \rightarrow q_4: \quad \delta(q_3, xxxxxx1) = q_4$$

$$\delta_5: q_4 \rightarrow q_5: \quad \delta(q_4, x1x0xx1) = q_5$$

Další možný způsob vyjádření těchto přechodů

$$\delta_1: q_0 \rightarrow q_1 (i_1 \cap i_2 \cap i_3 \cap i_4)$$

$$\delta_2: q_1 \rightarrow q_2 (i_5)$$

$$\delta_3: q_2 \rightarrow q_3 (i_6)$$

$$\delta_4: q_3 \rightarrow q_4 (i_7)$$

$$\delta_5: q_4 \rightarrow q_5 (i_2 \cap \bar{i}_4 \cap i_7)$$

Vyjádření přechodových stavů tabulkou

$Q \backslash \Sigma^2$	i_1	i_2	i_3	i_4	i_5	i_6	i_7	Σ
$\delta_1: q_0 \rightarrow q_1$	1	1	1	1	x	x	x	a
$\delta_2: q_1 \rightarrow q_2$	x	x	x	x	1	x	x	b
$\delta_3: q_2 \rightarrow q_3$	x	x	x	x	x	1	x	c
$\delta_4: q_3 \rightarrow q_4$	x	x	x	x	x	x	1	d
$\delta_5: q_4 \rightarrow q_5$	x	1	x	0	x	x	1	e

Obr. 14 Tabulka přechodových stavů - výroba produktu A

V tabulce na *Obr. 14* zatím neuvažuji s přechody do poruchového stavu $\delta_E: q_i \rightarrow q_E$, ty doplním později.

Vstupní abeceda Σ odvozená od vstupů $\Sigma^\Sigma = \{ i_1, i_2, i_3, i_4, i_5, i_6, i_7 \}$ je patrná z výše uvedené tabulky. Pro vstupní abecedu platí, že:

$$a = (i_1 \cap i_2 \cap i_3 \cap i_4)$$

$$b = (i_5)$$

$$c = (i_6)$$

$$d = (i_7)$$

$$e = (i_2 \cap \bar{i}_4 \cap i_7)$$

Potom tedy platí, že vstupní abeceda je $\Sigma = \{ a, b, c, d, e \}$

Výchozí stav technologického procesu q_0 :

Jedná se o výchozí stav, kdy je technologie připravena k zahájení činnosti. Pro uvedený příklad je počáteční stav definován, jako stav, kdy jsou aktivní vstupy i_2, i_3, i_4 , tedy teplota je menší než ϑ_1 , reaktor je prázdný a vypouštěcí ventil V3 je uzavřen. Za této situace je technologický proces ve výchozím stavu q_0 ve kterém může setrvat bez jakéhokoliv časového omezení.

Koncový stav technologického procesu F :

Dle definice $F \subseteq Q$ tedy v tomto případě $F = q_5$

Stav při chybě/poruše technologického procesu E :

Opět vyjdu z definice kde $E \subseteq Q$ tedy v tomto případě $E = q_E$. Kde stav q_E definuji jako společný pro všechny diagnostikované uzly, v případně požadavku lze pro každý diagnostikovaný uzel definovat vlastní poruchový stav. Pro naše účely je však dostačující, pokud budeme předpokládat jeden společný poruchový stav, který bude do řídicího automatu signalizovat „Porucha procesu“. Tyto přechody budou aktivovány od překročení maximálního času přechodu δ_i , nebo stavu q_i .

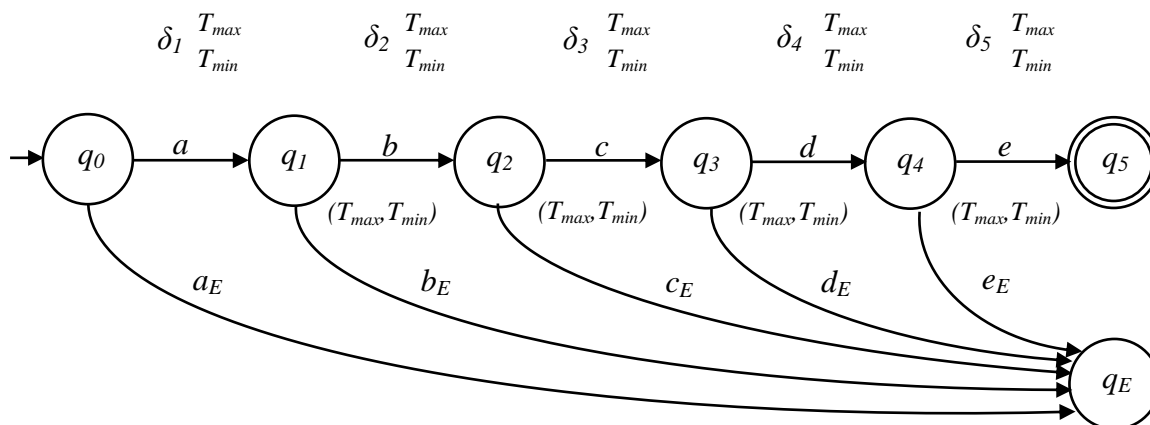
Nyní tabulku přechodů doplním, o přechody do poruchového stavu, tím získám přechodovou tabulku, která je uvedena na *Obr. 15*. Přechody δ_i jsou shodné s přechody v tabulce na *Obr. 14*. Přechody δ_{iE} do poruchového stavu q_E jsou aktivovány od maximálního času doby přechody, nebo doby trvání stavu.

přechod vstup	i_1	i_2	i_3	i_4	i_5	i_6	i_7	$(t_\delta \geq T_{\max}) \text{ or } (t_q \geq T_{\max})$	Σ
$\delta_1: q_o \rightarrow q_1$	1	1	1	1	x	x	x	0	a
$\delta_{1E}: q_o \rightarrow q_E$	1	1	1	1	x	x	x	1	a_E
$\delta_2: q_1 \rightarrow q_2$	x	x	x	x	1	x	x	0	b
$\delta_{2E}: q_1 \rightarrow q_E$	x	x	x	x	1	x	x	1	b_E
$\delta_3: q_2 \rightarrow q_3$	x	x	x	x	x	1	x	0	c
$\delta_{3E}: q_2 \rightarrow q_E$	x	x	x	x	x	1	x	1	c_E
$\delta_4: q_3 \rightarrow q_4$	x	x	x	x	x	x	1	0	d
$\delta_{4E}: q_3 \rightarrow q_E$	x	x	x	x	x	x	1	1	d_E
$\delta_5: q_4 \rightarrow q_5$	x	1	x	0	x	x	1	0	e
$\delta_{5E}: q_4 \rightarrow q_E$	x	x	x	x	x	x	1	1	e_E

Obr. 15 Přechodová tabulka doplněná o přechody do poruchových stavů

Tímto jsem nadefinoval osmici A_{pd} konečného automatu pro predikci, kde vstupní abeceda Σ , včetně poruchových stavů, je dána konečnou množinou událostí podle tabulky uvedené na *Obr. 15* jako $\Sigma = \{ a, a_E, b, b_E, c, c_E, d, d_E, e, e_E \}$.

Nyní přistoupím ke stanovení časových řad α_i , podle (3.18). Tyto časové řady jsou zásobníky dat pro predikci chování technologického procesu. Nejprve určím stavy a přechody, tedy prediktivní uzly, na základě kterých bude možné provádět predikci technologického procesu. K určení diagnostikovaných uzlů je dobré sestavit stavový diagram a na něm určit příslušné přechody a stavy, které jsou rozhodující pro diagnostiku. Stavový diagram sledovaného technologického procesu je uveden na *Obr. 16*, stav při poruše q_E je uvažován jeden společný.



Obr. 16 Stavový diagram technologického procesu – výroba produktu A

Časové řady α :

Obecně lze konstatovat, že jsou k dispozici časové řady pro všechny stavy:

α_{q_0} pro stav q_0

α_{q_1} pro stav q_1

α_{q_2} pro stav q_2

α_{q_3} pro stav q_3

α_{q_4} pro stav q_4

α_{q_5} pro stav q_5

stejně tak i pro všechny přechody:

$\alpha_{q_0 \rightarrow q_1}$ pro přechod ze stavu q_0 do stavu q_1

$\alpha_{q_1 \rightarrow q_2}$ pro přechod ze stavu q_1 do stavu q_2

$\alpha_{q_2 \rightarrow q_3}$ pro přechod ze stavu q_2 do stavu q_3

$\alpha_{q_3 \rightarrow q_4}$ pro přechod ze stavu q_3 do stavu q_4

$\alpha_{q_4 \rightarrow q_5}$ pro přechod ze stavu q_4 do stavu q_5

Máme tedy k dispozici tyto časové řady:

$$\alpha_{q_0} \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

$$\alpha_{q_0 \rightarrow q_1} = \delta_1 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

$$\alpha_{q_1} \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

$$\alpha_{q_1 \rightarrow q_2} = \delta_2 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

$$\alpha_{q_2} \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

$$\alpha_{q_2 \rightarrow q_3} = \delta_3 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

$$\alpha_{q_3} \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

$$\alpha_{q_3 \rightarrow q_4} = \delta_4 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \};$$

$$\alpha_{q_4} \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

$$\alpha_{q_4 \rightarrow q_5} = \delta_5 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \};$$

$$\alpha_{q_5} \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \}$$

Při bližším prozkoumání uvedených stavů a přechodů, je zřejmé, že ne všechny budou mít pro reálný technologický proces nějaký význam. Jak jsem se již dříve zmínil, tak výchozí stav q_0 a stejně tak i koncový stav q_5 mohou trvat neomezeně dlouhou dobu, to znamená, že časové řady α_{q_0} a α_{q_5} není potřeba, pro účely diagnostiky, nijak sledovat co se délky jejich trvání týká. To lze konstatovat u technologických procesů obecně. Ostatní stavy mají pro technologický proces smysl. Přechody, mimo přechodů při poruše, mají smysl všechny.

Výše uvedené skutečnosti je možné pro technologické procesy zevšeobecnit a potom mohu konstatovat, že jako diagnostikovaný uzel lze určit sledování, jak dlouho trvají jednotlivé stavy technologického procesu mimo stavu počátečního a koncového. To mohu vyjádřit takto:

$$q_i \in Q \text{ pro } i \in \{1; n - 1\} \quad (3.21)$$

kde n je počet stavů nad q_0 .

Dále mohu jako diagnostikovaný uzel určit všechny přechody mezi stavy $\alpha_{q_0 \rightarrow q_1} = \delta_1$; $\alpha_{q_1 \rightarrow q_2} = \delta_2$; $\alpha_{q_2 \rightarrow q_3} = \delta_3$; $\alpha_{q_3 \rightarrow q_4} = \delta_4$; $\alpha_{q_4 \rightarrow q_5} = \delta_5$. Toto tvrzení mohu opět zevšeobecnit a to do vztahu:

$$\alpha_{(q_{i-1}) \rightarrow q_i} = \delta_i \text{ pro } i \in \langle 1; n \rangle \quad (3.22)$$

kde n je počet stavů nad q_0 .

Tímto mám určeny diagnostikované uzly, které jsou, s přihlédnutím k (3.21) a (3.22), pro uvedený příklad, následující:

$$\begin{aligned} \alpha_{q_0 \rightarrow q_1} &= \delta_1 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \} \\ \alpha_{q_1} & \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \} \\ \alpha_{q_1 \rightarrow q_2} &= \delta_2 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \} \\ \alpha_{q_2} & \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \} \\ \alpha_{q_2 \rightarrow q_3} &= \delta_3 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \} \\ \alpha_{q_3} & \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \} \\ \alpha_{q_3 \rightarrow q_4} &= \delta_4 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \} \\ \alpha_{q_4} & \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \} \\ \alpha_{q_4 \rightarrow q_5} &= \delta_5 \{ \alpha_{T_0}; \alpha_{T_1}; \alpha_{T_2}; \dots \alpha_{T_n} \} \end{aligned} \quad (3.23)$$

Počet diagnostikovaných uzlů odpovídá počtu přechodů (3.22) a počtu stavů mimo počátečního stavu q_0 a koncového stavu q_n (3.21). Počet diagnostikovaných uzlů α_T , ve kterých je účelné provádět kontrolu trvání stavu/přechodu lze tedy obecně vyjádřit matematicky jako:

$$\alpha_{TC} = (2s - 3) \quad (3.24)$$

kde α_{TC} je počet uzlů diagnostikovaných časově

s je počet stavů technologického procesu

V uvedeném příkladu je počet stavů technologického procesu $s = 6$. Jsou to stavy $q_0, q_1, q_2, q_3, q_4, q_5$. Počet časově diagnostikovaných uzlů dle (3.24) je 9 a jsou uvedeny ve vztahu (3.23).

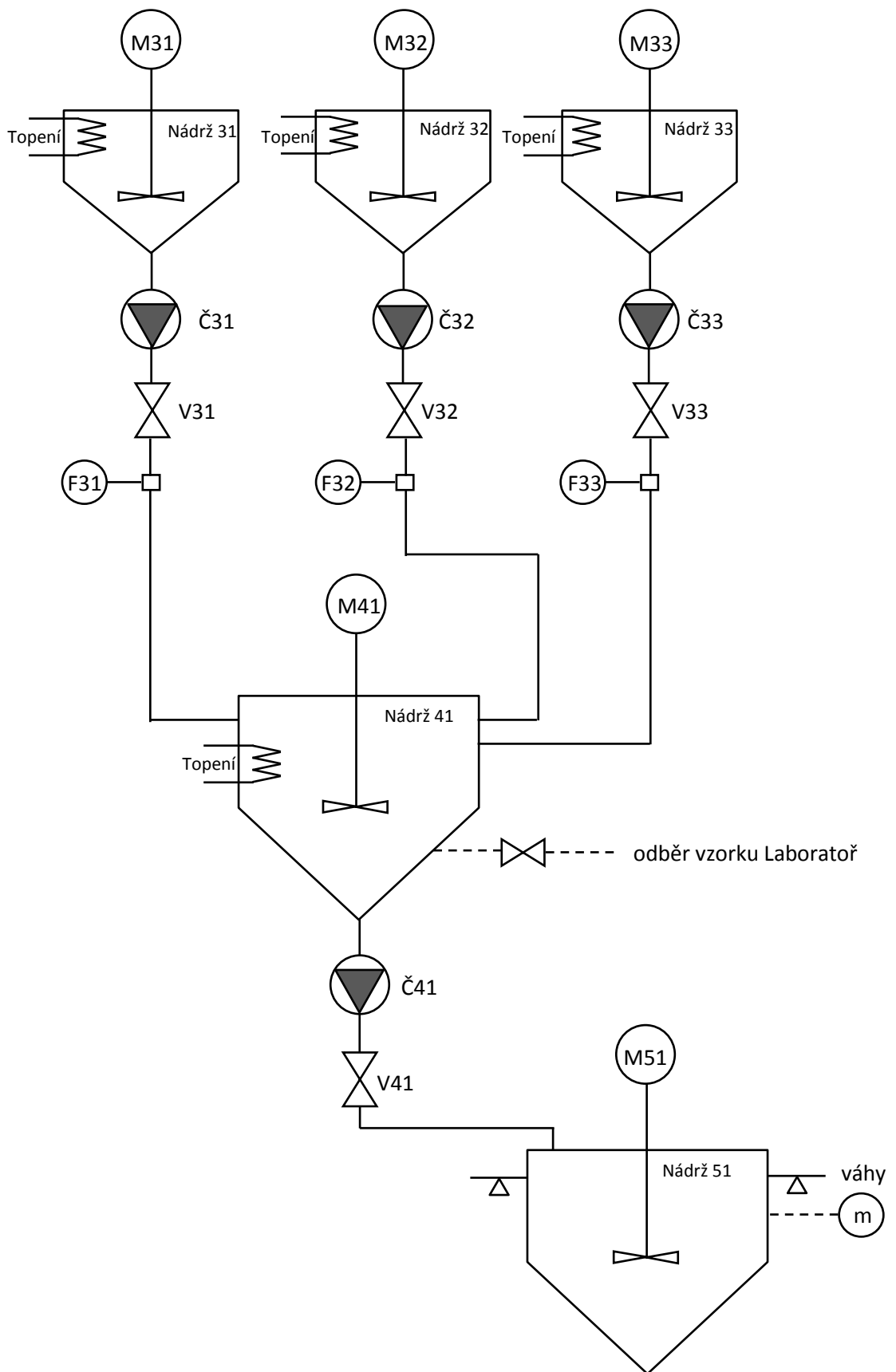
Tímto jsem nadefinoval parametry konečného automatu pro prediktivní diagnostiku s tím, že časy T_{min}, T_{max} je třeba určit ze zadání technologického procesu, nebo měřením z reálného procesu. V kapitole 3.4.2.1 jsou uvedeny výstupy pro tento vzorový příklad.

Na *Obr. 20* je uvedeno blokové schéma prediktivní diagnostiky technologického procesu. Z uvedeného obrázku je patrné, že vlastní technologický proces je řízen řídicím systémem, který předává informace do prediktivního diagnostického modelu daného procesu, se kterým pracuje paralelně. Model tedy v každém kroku pracuje s reálnými údaji získanými měřeními a předzpracovanými řídicím systémem do tvaru pro potřeby konečného automatu pro prediktivní diagnostiku. Mohou nastat i případy, kdy je technologický proces řízen konečným automatem a potom může být řídicí konečný automat zároveň využit jako konečný prediktivní automat pro diagnostiku. V těchto případech je použit konečný automat s výstupem.

Dále je možné, že v případech kdy je technologický proces řízen vlastním systémem a konečný automat je využit pouze jako model pro prediktivní diagnostiku, lze model technologického procesu realizovat také jako konečný automat s výstupem, který je vstupem do řídicího systému z diagnostiky. Tento výstup z diagnostiky může fungovat jako nadřazená ochrana technologického procesu.

3.4.1.2. Příklad technologického procesu 2 – Výroba maziva mícháním

Na obrázku *Obr. 17* je uveden zjednodušený příklad technologického procesu výroby průmyslového maziva mícháním olejů. Technologie míchání olejů spočívá v dokonalém nadávkování a smíchání jednotlivých komponent (olejů a aditiv) v poměrech podle zadaných receptur. Celý technologický proces se skládá z nádrží, čerpadel, míchadel, topení a ventilů.



Obr. 17 Příklad technologického procesu – výroba produktu maziva mícháním olejů

System pracuje na základě spuštění požadované receptury, která spočívá v zadání typu a množství požadovaných aditiv. Na *Obr. 17* je znázorněno technologické zařízení přímo potřebné pro recepturu, kterou nazvu **R1**. Receptura probíhá tak, že pomocí čerpadel Č31 až Č32, přes ventily V31 až V32 se nadávkuje požadované množství základového oleje z nádrže 31 a aditiv z nádrží 32 a 33 do nádrže 41. V nádrži 41 je udržována požadovaná teplota a zároveň dochází k promíchávání aditiv po stanovenou dobu. Pro uvedený příklad budu předpokládat, že napouštění oleje a aditiv probíhá současně a pro diagnostiku nebudu kontrolovat jednotlivé větve média, jako je tomu v předchozím příkladu uvedeném na *Obr. 13*. Průběžně se provádí ruční odběr vzorku a v laboratoři je realizována kontrola kvality konečného produktu. Po dosažení požadovaných parametrů je konečný produkt pomocí čerpadla Č41 přes ventil V41 přepuštěn do nádrže 51. Nádrž 51 je vybavena tenzometrickými váhami pro kontrolu množství konečného produktu.

Slovní popis receptury R1

Po zadání povelu na zahájení výroby, je pro danou recepturu provedena kontrola dostatečného množství základového oleje v nádrži 31 a aditiv v nádržích 32 a 33, pomocí měření hladiny v nádržích (L31 až L33). Jsou-li všechny hladiny větší než $L_{minprov}$, je zahájeno dávkování základového oleje a aditiv (otevrou ventily V31 až V33 a zároveň se spustí čerpadla Č31 až Č33). Zároveň je automaticky spuštěno topení a míchadlo v nádrži M41. Hmotnostními průtokoměry F31 až F33 je měřeno požadované množství dávkovaného oleje a příslušných aditiv. Po dosažení požadovaného množství (F_{31a} , F_{32a} , F_{33a}) materiálu vypnou příslušná čerpadla a zavřou příslušné ventily. Po nadávkování základového oleje a všech aditiv je, za stálého chodu míchadla M41, udržována v nádrži 41 konstantní teplota ϑ_K . Po uplynutí času T je proveden odběr vzorku, pokud je vzorek označen jako vyhovující, je materiál přepuštěn z nádrže 41 přes ventil V41, čerpadlem Č41 do nádrže 51. Při zahájení přepouštění v nádrži 51 sepne míchadlo M51. Finální množství materiálu v nádrži 51 je kontrolováno měřením hmotnosti.

3.4.1.2.1. Tvorba modelu

Stejně jako u předchozího příkladu uvedeném v odstavci 3.4.1.1 i v tomto případě vytvořím model technologického procesu pomocí konečného automatu pro diagnostiku technologického procesu definovaném v odstavci 3.3.4.

Podle vztahů (3.19) a (3.20) máme:

$$A_{pd} = (Q, \Sigma, T_{min}, T_{max}, \delta, q_0, F, E)$$

$$\alpha_i = \{\alpha_{Ti}\}$$

Opět nejdříve určím jednotlivé množiny, technologického procesu, které vstupují do modelu. Zde je opět na místě připomenout, že tvořím model technologického procesu, vlastní řízení procesu je realizováno řídicím systémem. Vstupy do modelu získám z řídicího systému technologického procesu.

Stavy technologického procesu Q:

- Proces je připraven stav q_0
- Probíhá napouštění nádrže 41 stav q_1
- Ukončení napouštění oleje/aditiv stav q_2
- Promíchávání materiálu stav q_3
- Odběr vzorku stav q_4
- Ukončení míchání stav q_5
- Přečerpání do zásobníku 51 stav q_6
- Porucha výroby stav q_E

Neprázná konečná množina stavů je potom $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_E\}$

Vstupy technologického procesu Σ^Z :

- spuštění receptury R1 vstup i_1
- Hladina v nádrži 31 $> L_{minprov}$ vstup i_2
- Hladina v nádrži 32 $> L_{minprov}$ vstup i_3
- Hladina v nádrži 33 $> L_{minprov}$ vstup i_4
- V31 zavřen: vstup i_5
- V32 zavřen: vstup i_6
- V33 zavřen: vstup i_7
- Chod čerpadla Č31 vstup i_8
- Chod čerpadla Č32 vstup i_9
- Chod čerpadla Č33 vstup i_{10}

- Množství materiálu F31= F_{31a} vstup i_{11}
- Množství materiálu F32= F_{32a} vstup i_{12}
- Množství materiálu F33= F_{33a} vstup i_{13}
- Zapnuto míchadlo 41 vstup i_{14}
- Zapnuto topení v nádrži 41 vstup i_{15}
- Teplota v nádrži 41 $\geq \vartheta_K$ vstup i_{16}
- čas T vstup i_{17}
- vzorek je vyhovující vstup i_{18}
- V41 zavřen vstup i_{19}
- Chod čerpadla Č41 vstup i_{20}

Události neboli vstupní abeceda Σ , je potom konečná množina událostí způsobená vstupy $\Sigma^\Sigma \{ i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10}, i_{11}, i_{12}, i_{13}, i_{14}, i_{15}, i_{16}, i_{17}, i_{18}, i_{19}, i_{20} \}$.

Množina časů T_{min}, T_{max} :

Minimální a maximální časy budou určeny pro jednotlivé přechody a stavy, pro které jsou smysluplné. Pro vzorový příklad budou časy opět určeny.

Podmínky pro přechod stavů δ :

Pro vstupy $\Sigma^\Sigma = \{ i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10}, i_{11}, i_{12}, i_{13}, i_{14}, i_{15}, i_{16}, i_{17}, i_{18}, i_{19}, i_{20} \}$ jsou níže uvedeny přechodové funkce pro přechody mezi stavy s tím, že x značí libovolnou hodnotu vstupu i_n

$$\delta_1: q_0 \rightarrow q_1: \quad \delta(q_0, 1xxx000111xxxxxxxxxx) = q_1$$

$$\delta_2: q_1 \rightarrow q_2: \quad \delta(q_1, xxxxxxxxxxx111xxxxxxxx) = q_2$$

$$\delta_3: q_2 \rightarrow q_3: \quad \delta(q_2, xxxx111000xxx111xxxx) = q_3$$

$$\delta_4: q_3 \rightarrow q_4: \quad \delta(q_3, xxxx111000xxxxxx1xxx) = q_4$$

$$\delta_5: q_4 \rightarrow q_5: \quad \delta(q_4, xxxx111000xxxxxxx1xx) = q_5$$

$$\delta_5: q_5 \rightarrow q_6: \quad \delta(q_5, xxxx111000xxxxxxxx11) = q_6$$

Další možný způsob vyjádření těchto přechodů

$$\delta_1: q_0 \rightarrow q_1 (i_1 \cap \bar{i}_5 \cap \bar{i}_6 \cap \bar{i}_7 \cap i_8 \cap i_9 \cap i_{10})$$

$$\delta_2: q_1 \rightarrow q_2 (i_{11} \cap i_{12} \cap i_{13})$$

$$\delta_3: q_2 \rightarrow q_3 (i_5 \cap i_6 \cap i_7 \cap \bar{i}_8 \cap \bar{i}_9 \cap \bar{i}_{10} \cap i_{14} \cap i_{15} \cap i_{16})$$

$$\delta_4: q_3 \rightarrow q_4 (i_5 \cap i_6 \cap i_7 \cap \bar{i}_8 \cap \bar{i}_9 \cap \bar{i}_{10} \cap i_{17})$$

$$\delta_5: q_4 \rightarrow q_5 (i_5 \cap \bar{i}_6 \cap i_7 \cap \bar{i}_8 \cap \bar{i}_9 \cap \bar{i}_{10} \cap i_{18})$$

$$\delta_6: q_4 \rightarrow q_5 (i_5 \cap \bar{i}_6 \cap i_7 \cap \bar{i}_8 \cap \bar{i}_9 \cap \bar{i}_{10} \cap i_{19} \cap i_{20})$$

Vyjádření přechodových stavů tabulkou je na následující stránce na obrázku *Obr. 18*

	l_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}	i_{11}	i_{12}	i_{13}	i_{14}	i_{15}	i_{16}	i_{17}	i_{18}	i_{19}	i_{20}		
$\delta_1: q_0 \rightarrow q_1$	1	x	x	x	0	0	0	1	1	1	x	x	x	x	x	x	x	x	x	x	x	a
$\delta_{1E}: q_0 \rightarrow q_E$	1	x	x	x	0	0	0	1	1	1	x	x	x	x	x	x	x	x	x	x	1	a _E
$\delta_2: q_1 \rightarrow q_2$	x	x	x	x	x	x	x	x	x	x	1	1	1	x	x	x	x	x	x	x	x	b
$\delta_{2E}: q_1 \rightarrow q_E$	x	x	x	x	x	x	x	x	x	x	1	1	1	x	x	x	x	x	x	x	1	b _E
$\delta_3: q_2 \rightarrow q_3$	x	x	x	x	1	1	1	0	0	0	x	x	x	1	1	1	x	x	x	x	x	c
$\delta_{3E}: q_2 \rightarrow q_E$	x	x	x	x	1	1	1	0	0	0	x	x	x	1	1	1	x	x	x	x	1	c _E
$\delta_4: q_3 \rightarrow q_4$	x	x	x	x	1	1	1	0	0	0	x	x	x	x	x	x	1	x	x	x	x	d
$\delta_{4E}: q_3 \rightarrow q_E$	x	x	x	x	1	1	1	0	0	0	x	x	x	x	x	x	1	x	x	x	1	d _E
$\delta_5: q_4 \rightarrow q_5$	x	x	x	x	1	1	1	0	0	0	x	x	x	x	x	x	x	1	x	x	x	e
$\delta_{5E}: q_4 \rightarrow q_E$	x	x	x	x	1	1	1	0	0	0	x	x	x	x	x	x	x	1	x	x	1	e _E
$\delta_6: q_5 \rightarrow q_6$	x	x	x	x	1	1	1	0	0	0	x	x	x	x	x	x	x	x	1	1	x	f
$\delta_{6E}: q_5 \rightarrow q_E$	x	x	x	x	1	1	1	0	0	0	x	x	x	x	x	x	x	x	1	1	1	f _E

Obr. 18 Tabulka přechodových stavů – výroba maziva

Z tabulky na *Obr. 18* vyplývá, že vstupní abeceda Σ je

$$\Sigma = \{ a, a_E, b, b_E, c, c_E, d, d_E, e, e_E, f, f_E \}$$

Výchozí stav technologického procesu q_0 :

Jedná se o výchozí stav, kdy je technologie připravena k zahájení činnosti. Pro tento příklad je počáteční stav definován, jako stav, kdy jsou aktivní vstupy $i_2, i_3, i_4, i_5, i_6, i_7$ tedy hladiny v nádržích 31, 32 a 33 jsou větší než $L_{minprov}$ a příslušné vypouštěcí ventily V31, V32 a V33 jsou zavřeny. Za této situace je technologický proces ve výchozím stavu q_0 ve kterém může setrvat bez jakéhokoliv časového omezení.

Koncový stav technologického procesu F :

Dle definice $F \subseteq Q$ tedy v tomto případě $F = q_6$

Stav při chybě/poruše technologického procesu E :

Poruchový stav $E \subseteq Q$, v tomto případě $E = q_E$. Kde stav q_E je definován jako společný pro všechny diagnostikované uzly, a který bude do řídicího systému signalizovat „Porucha procesu“. Tyto přechody budou aktivovány od překročení maximálních časů přechodu δ_i , nebo stavu q_i .

Tímto jsem nadefinoval všechny množiny (osmice A_{pd}), konečného automatu pro predikci, se vstupní abecedou definovanou podle tabulky na *Obr. 18* jako:

$$\Sigma = \{ a, a_E, b, b_E, c, c_E, d, d_E, e, e_E, f, f_E \}$$

Nyní definuji časové řady α_i , pro predikci chování technologického procesu. Nejdříve určím prediktivní uzly, pro predikci technologického procesu. Pro určení diagnostikovaných uzlů sestrojím stavový diagram a na něm určím příslušné přechody a stavy, které jsou rozhodující pro diagnostiku. Stavový diagram sledovaného technologického procesu je uveden na *Obr. 19*, stav při poruše q_E uvažuji jeden společný. Diagnostikované uzly pro stavy α_{q_i} určím podle vztahu (3.21) a pro přechody $\alpha_{(q_{i-1}) \rightarrow q_i}$ podle vztahu (3.22). Počet stavů nad $q_0, n = 6$. Potom je tedy:

$$\alpha_{q_i} \in Q \text{ pro } i \in \{1; 5\}$$

$$\alpha_{(q_{i-1}) \rightarrow q_i} = \delta_i \text{ pro } i \in \{1; 6\}$$

Celkový počet časově diagnostikovaných uzlů α_{TC} určím podle vztahu (3.24), s tím, že počet stavů technologického procesu podle příkladu v kapitole 3.4.1.2 je $s = 7$. Potom je tedy:

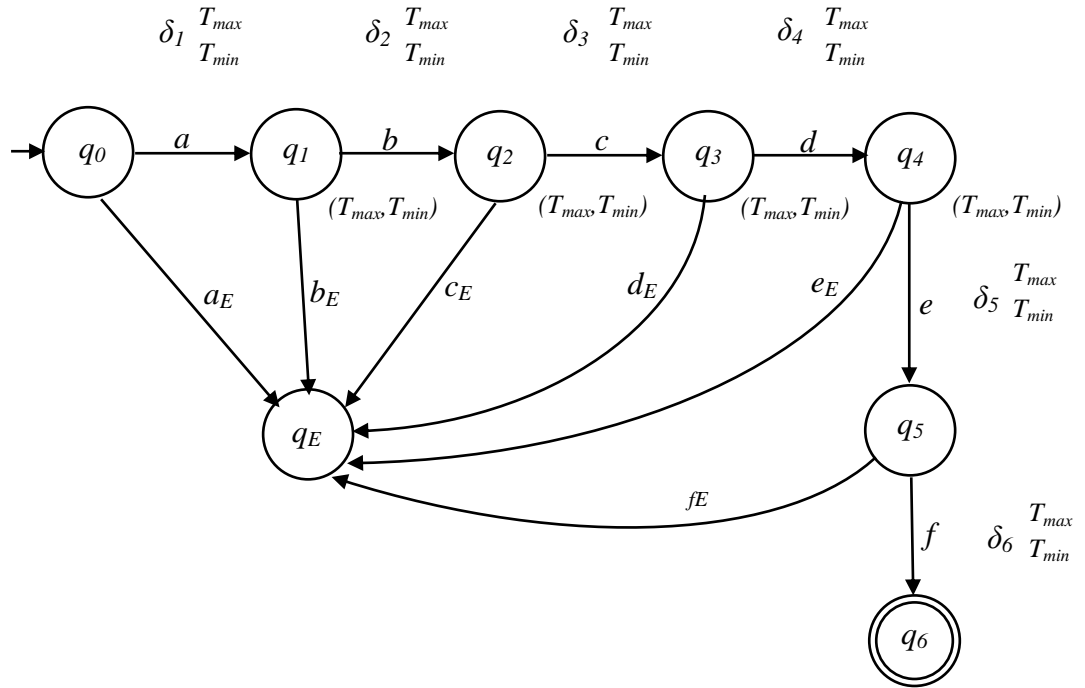
$$\alpha_{TC} = (2s - 3) = 11$$

Časové řady α :

$$\begin{aligned}
 \alpha_{q0 \rightarrow q1} &= \delta_1 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\
 \alpha_{q1} & \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\
 \alpha_{q1 \rightarrow q2} &= \delta_2 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\
 \alpha_{q2} & \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\
 \alpha_{q2 \rightarrow q3} &= \delta_3 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\
 \alpha_{q3} & \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\
 \alpha_{q3 \rightarrow q4} &= \delta_4 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \}; \\
 \alpha_{q4} & \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\
 \alpha_{q4 \rightarrow q5} &= \delta_5 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \}; \\
 \alpha_{q5} & \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\
 \alpha_{q5 \rightarrow q6} &= \delta_6 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \};
 \end{aligned} \tag{3.25}$$

Tím máme definovány parametry konečného automatu pro prediktivní diagnostiku s tím, že časy T_{min} , T_{max} , jsou určeny ze zadání technologického procesu, nebo měřením z reálného procesu.

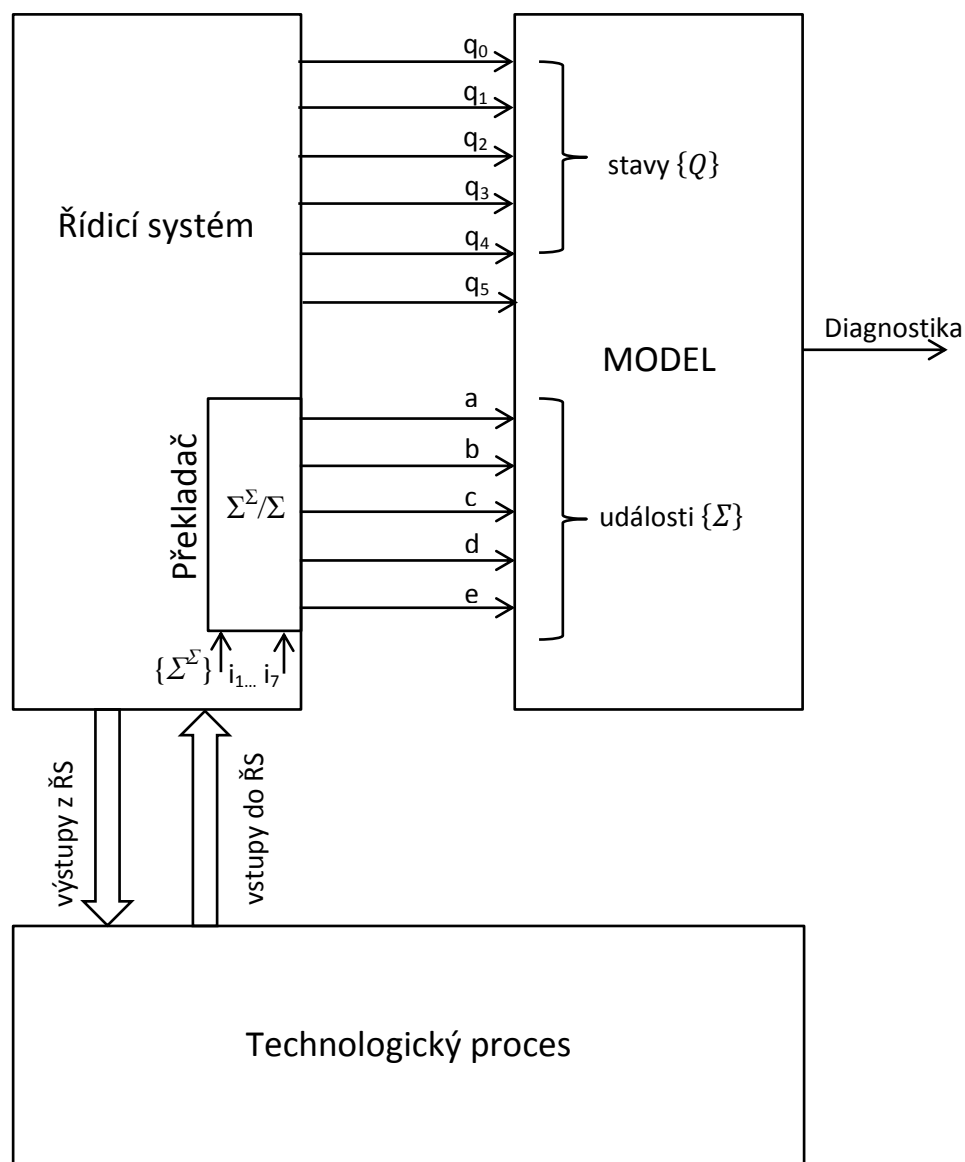
Blokové schéma prediktivní diagnostiky technologického procesu modelem je znázorněno na *Obr. 20*. Kde je znázorněna vazba mezi Technologickým procesem, Řídicím systémem a Modelem technologického procesu. Dále je zde znázorněn tok dat pro diagnostiku technologického procesu.



Obr. 19 Stavový diagram technologického procesu - výroba maziva

3.4.2. Funkce modelu

Model technologického procesu vytvořený konečným automatem pro diagnostiku získává své vstupy zřídícího systému, který technologický proces řídí, viz *Obr. 20*.



Obr. 20 Prediktivní diagnostika technologického procesu – blokové schéma

Model technologického procesu představuje osmici, podle vztahu (3.12) :

$$A_{pd} = (Q, \Sigma, T_{min}, T_{max}, \delta, q_0, F, E)$$

a časové řady:

$$\alpha_i = \{\alpha_{Ti}\}$$

3.4.2.1. Parametry a výstupy modelu podle Příkladu 1

Parametry modelu byly nadefinovány v odstavci 3.4.1.1.1 model je potom:

$$A_{pd} = ((q_0, q_1, q_2, q_3, q_4, q_5, q_E), (a, a_E, b, b_E, c, c_E, d, d_E, e, e_E) T_{min}, T_{max}, \delta, q_0, q_5, q_E) \quad (3.26)$$

$$\alpha_{q_0 \rightarrow q_1} = \delta_1 \{\alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn}\}$$

$$\alpha_{q_1} \{\alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn}\}$$

$$\alpha_{q_1 \rightarrow q_2} = \delta_2 \{\alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn}\}$$

$$\alpha_{q_2} \{\alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn}\}$$

(3.27)

$$\alpha_{q_2 \rightarrow q_3} = \delta_3 \{\alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn}\}$$

$$\alpha_{q_3} \{\alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn}\}$$

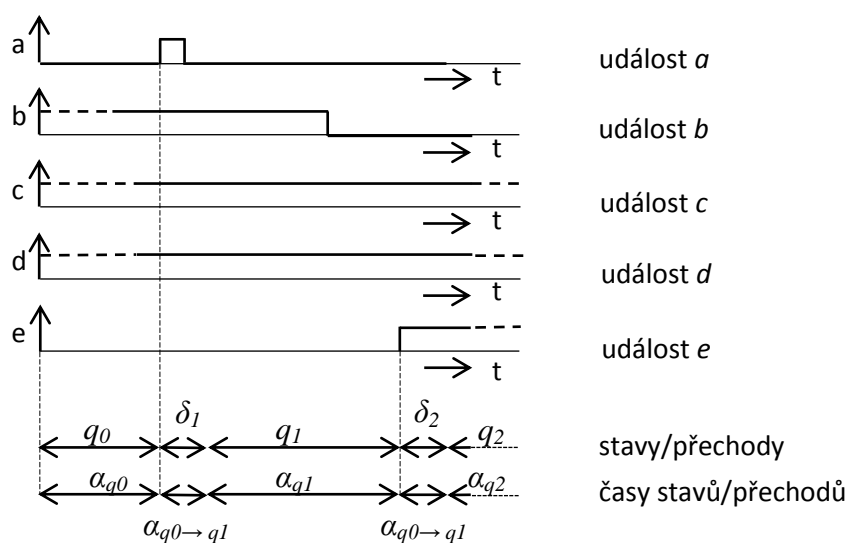
$$\alpha_{q_3 \rightarrow q_4} = \delta_4 \{\alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn}\}$$

$$\alpha_{q_4} \{\alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn}\}$$

$$\alpha_{q_4 \rightarrow q_5} = \delta_5 \{\alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn}\}$$

Model pracuje paralelně s vlastním řídicím systémem technologického procesu, vstupem do modelu jsou pak data v reálném čase. Model zpracovává a archivuje tato aktuální data o stavech a přechodech technologického procesu získaná z řídicího systému. Na základě získaných údajů, ukládaných ve formě časových řad α_i , má model přehled o současných a minulých hodnotách času trvání příslušných přechodů mezi stavy a trvání stavů. Časové řady α_i jsou při prvním spuštění prázdné a plní se údaji postupně při provozu. Jak je při prvním spuštění řada α_i nulová, tak se při provozu plní údaji teoreticky až do nekonečna, což

je nereálné a z technického hlediska zbytečné. Je proto třeba určit horizont predikce, tedy jak daleko dopředu budu chování technologického procesu predikovat a také je třeba stanovit rozsah časových řad α_i , tedy kolik prvků bude obsahovat (nebo jinak, s jak moc historickými daty budu pracovat). Ke sběru dat dochází průběžně, vždy mám k dispozici aktuální a historická data, ta můžu pro lepší využití koncentrovat, např. tak, že historicky nejstarší data budu umazávat, případně průměrovat údaje za určitá časová období čímž získáme historická data daleko do minulosti.



Obr. 21 Časová závislost přechodů a událostí

Na Obr. 21 je znázorněna časová závislost stavů a přechodů v technologickém procesu. Z obrázku je patrné, že jsou-li zároveň splněny události a, b, c, d, iniciuje se přechod ze stavu q_0 do stavu q_1 , délka trvání přechodu δ_1 je $\alpha_{q_0 \rightarrow q_1}$, délka trvání stavu q_1 je α_{q_1} . Po splnění události začne přechod ze stavu q_1 do stavu q_2 , délka trvání přechodu δ_2 je $\alpha_{q_1 \rightarrow q_2}$, délka trvání stavu q_2 je α_{q_2} . Pro úplnost je znázorněna i doba trvání stavu q_0 , tento stav ale může trvat neomezeně dlouho, nemá tedy pro diagnostiku praktický význam. Právě délka přechodu mezi jednotlivými stavy a doba trvání stavu jsou údaje významné pro diagnostiku, protože jak jsem již dříve uvedl a jak vyplývá z funkce technologického procesu, tyto časy by měly být při standardním, bezporuchovém stavu pro daný přechod, nebo stav vždy shodné.

Nejdůležitějšími daty pro prediktivní diagnostiku jsou časové řady $\alpha_i = \{\alpha_{T_i}\}$, které mají vypovídací hodnotu o chování technologickém procesu. Pro vzorový příklad, uvedený v odstavci 3.4.1.1 to jsou řady uvedené v (3.27) jsou to tedy $\alpha_{q_0 \rightarrow q_1}$, α_{q_1} , $\alpha_{q_1 \rightarrow q_2}$, α_{q_2} , $\alpha_{q_2 \rightarrow q_3}$, α_{q_3} , $\alpha_{q_3 \rightarrow q_4}$, α_{q_4} , $\alpha_{q_4 \rightarrow q_5}$. Při provozu se tyto řady plní údaji tj. doba trvání příslušného stavu/přechodu v jednotkách času [sec]. Časy pro vzorový příklad byly určeny.

Data vzorového příkladu:

$$\alpha_{q_0 \rightarrow q_1} = (40; 41; 40; 42; 40; 38; 41; 39; 42; 40)$$

$$\alpha_{q_1} = (285; 290; 280; 295; 285; 280; 285; 290; 285; 288)$$

$$\alpha_{q_1 \rightarrow q_2} = (68; 70; 75; 69; 75; 68; 70; 72; 69; 71)$$

$$\alpha_{q_2} = (325; 335; 328; 330; 335; 326; 335; 325; 330; 328)$$

$$\alpha_{q_2 \rightarrow q_3} = (80; 85; 90; 80; 85; 79; 85; 80; 85; 80)$$

$$\alpha_{q_3} = (95; 98; 85; 95; 87; 89; 94; 88; 90; 94)$$

$$\alpha_{q_3 \rightarrow q_4} = (40; 41; 40; 42; 40; 38; 41; 39; 42; 40)$$

$$\alpha_{q_4} = (65; 68; 67; 65; 68; 65; 64; 65; 64; 65)$$

$$\alpha_{q_4 \rightarrow q_5} = (73; 71; 72; 72; 71; 73; 72; 73; 72; 72)$$

Průběhy výše uvedených časových řad, jsou znázorněny v grafech, uvedených na *Obr. 22* až *Obr. 30*. Na těchto grafech je vždy uveden průběh příslušné časové řady a predikovaná hodnota. Budoucí (predikovaná) hodnota se ze změřených hodnot určí některou z predikčních metod viz. odstavec 3.1. V uvedených grafech jsem predikci realizoval metodou linearizace, kdy průběh historických hodnot linearizujeme a prodloužením získaného trendu můžeme určit pravděpodobné chování technologického procesu v horizontu predikce. Trend chování je pro predikci chování v technologickém procesu dostatečně vypovídajícím údajem, kterým lze odhalit změny/zhoršování stavu technologického zařízení, které je příčinou změny chování sledovaného technologického procesu. Na těchto grafech osa y reprezentuje dobu trvání stavu/přechodu v časových jednotkách, osa x představuje počty měření, s tím, že nezáleží na délce časového úseku mezi jednotlivými α_{T_i} . Ty mohou být dokonce různé bez vlivu na výsledky diagnostiky. Výhodou tohoto řešení je, že provádím průběžný sběr dat α_{T_i}

a historická data mohu koncentrovat (např. průměr, medián,...). Mohu tedy pracovat např. s deseti hodnotami α_{Ti} , kdy posledních pět hodnot (α_{T6} až α_{T10}) bude průběžných a každá z prvních pěti hodnot (α_{T1} až α_{T5}), bude představovat koncentrovaná data vždy po pěti hodnotách za použití mediánu. Ve skutečnosti potom tedy řada (α_{T1} až α_{T10}) představuje 30 hodnot.

Například:

$$\alpha_{T1} = \text{Me}\{ 40; 41; 43; 42; 40 \} = 41$$

$$\alpha_{T2} = \text{Me}\{ 43; 43; 45; 42; 40 \} = 43$$

$$\alpha_{T3} = \text{Me}\{ 42; 40; 44; 43; 40 \} = 42$$

$$\alpha_{T4} = \text{Me}\{ 44; 40; 41; 42; 40 \} = 41$$

$$\alpha_{T5} = \text{Me}\{ 41; 42; 43; 41; 42 \} = 42$$

$$\alpha_{T6} = 42$$

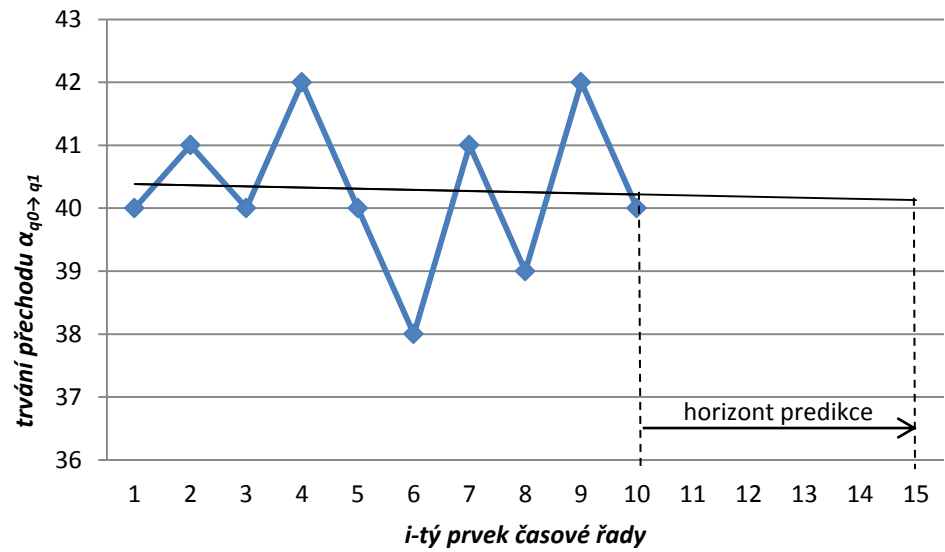
$$\alpha_{T7} = 43$$

$$\alpha_{T8} = 42$$

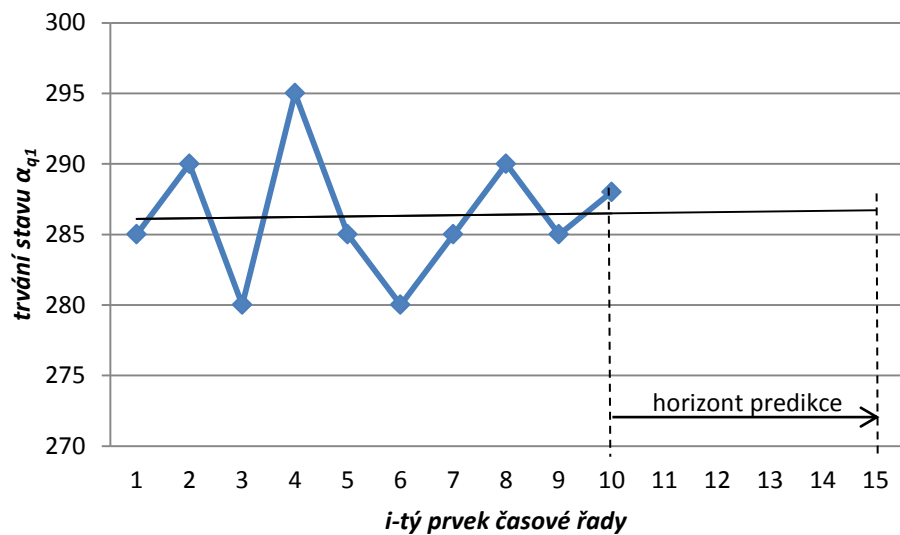
$$\alpha_{T9} = 42$$

$$\alpha_{T10} = 43$$

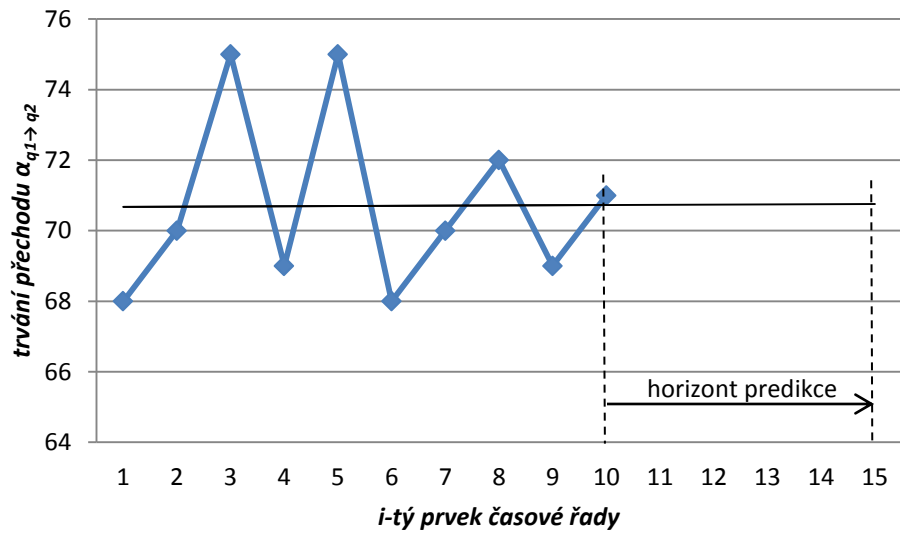
Potom je tedy α_{Ti} (41; 43; 42; 41; 42; 42; 43; 42; 42; 43)



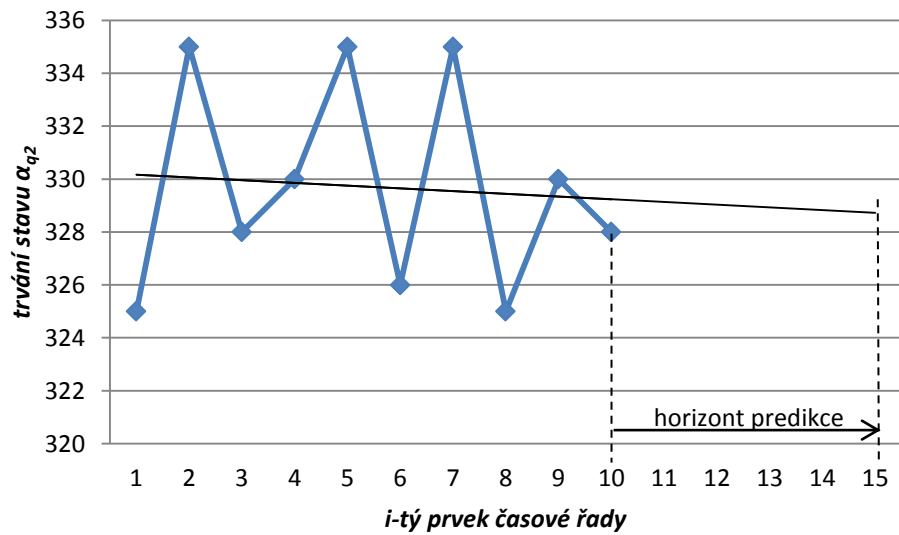
Obr. 22 Časová řada průběhu přechodu $\alpha_{q_0 \rightarrow q_1}$, a její linearizace



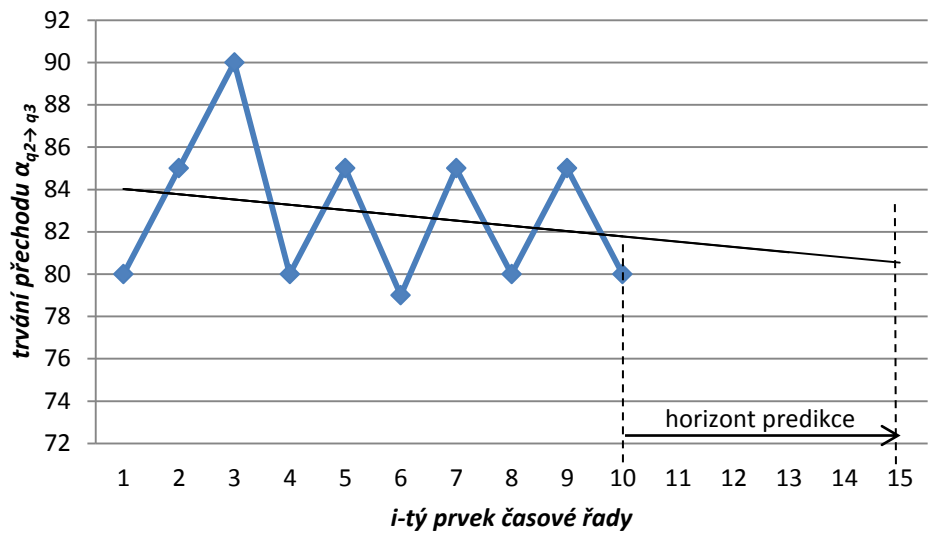
Obr. 23 Časová řada stavu α_{q_1} , a její linearizace



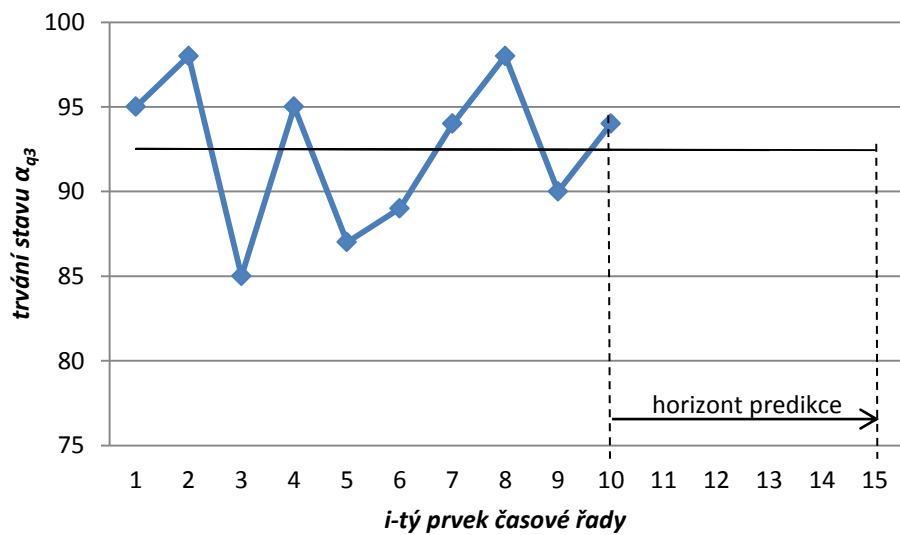
Obr. 24 Časová řada průběhu přechodu $\alpha_{q_1 \rightarrow q_2}$, a její linearizace



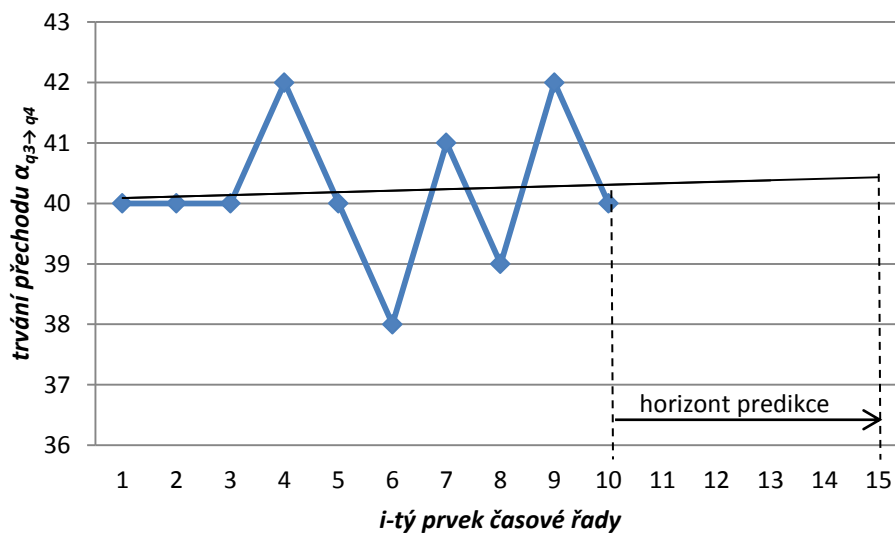
Obr. 25 Časová řada stavu α_{q_2} , a její linearizace



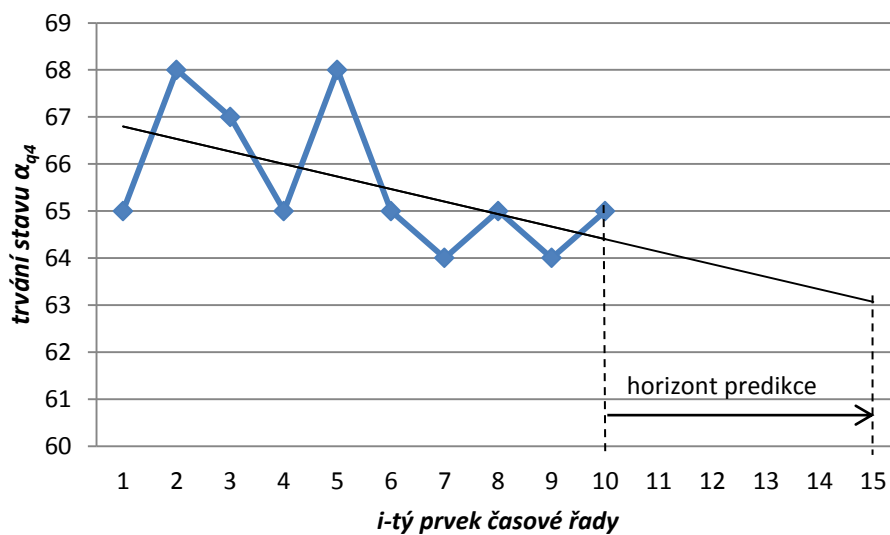
Obr. 26 Časová řada průběhu přechodu $\alpha_{q_2 \rightarrow q_3}$, a její linearizace



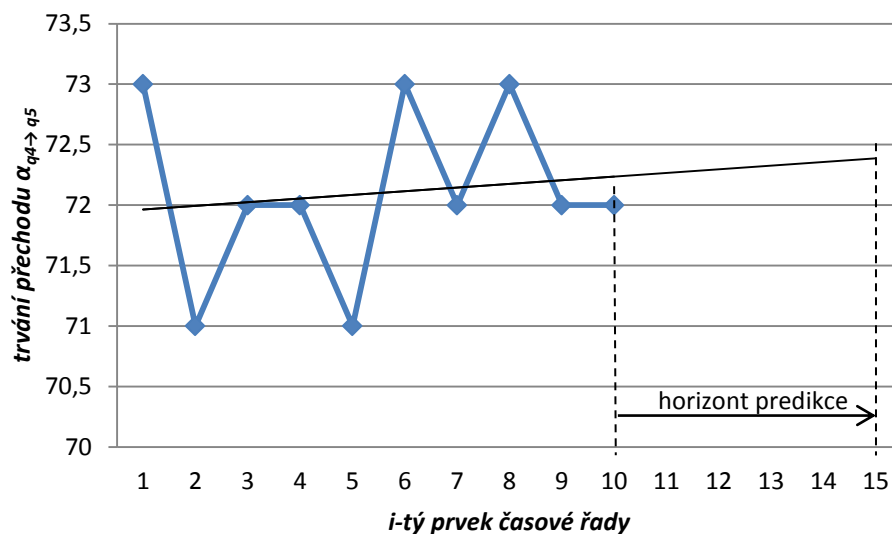
Obr. 27 Časová řada stavu α_{q_3} , a její linearizace



Obr. 28 Časová řada průběhu přechodu $\alpha_{q_3 \rightarrow q_4}$, a její linearizace



Obr. 29 Časová řada stavu α_{q_4} , a její linearizace



Obr. 30 Časová řada průběhu přechodu $\alpha_{q4 \rightarrow q5}$, a její linearizace

3.4.2.2. Parametry a výstupy modelu podle Příkladu 2

V tomto odstavci ještě stručně představím výstupy z druhého modelu, jehož parametry byly nadefinovány v odstavci 3.4.1.2. Model je představen následujícími vztahy (3.28) a (3.29).

$$A_{pd} = ((q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_E), (a, a_E, b, b_E, c, c_E, d, d_E, e, e_E, f, f_E) T_{min}, T_{max}, \delta, q_0, q_5, q_E) \quad (3.28)$$

$$\begin{aligned} \alpha_{q_0 \rightarrow q_1} &= \delta_1 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\ \alpha_{q_1} & \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\ \alpha_{q_1 \rightarrow q_2} &= \delta_2 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\ \alpha_{q_2} & \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\ \alpha_{q_2 \rightarrow q_3} &= \delta_3 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \\ \alpha_{q_3} & \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \} \end{aligned} \quad (3.29)$$

$$\alpha_{q3 \rightarrow q4} = \delta_4 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \}$$

$$\alpha_{q4} \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \}$$

$$\alpha_{q4 \rightarrow q5} = \delta_5 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \}$$

$$\alpha_{q5} \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \}$$

$$\alpha_{q5 \rightarrow q6} = \delta_6 \{ \alpha_{T0}; \alpha_{T1}; \alpha_{T2}; \dots \alpha_{Tn} \}$$

Jako parametry časových řad (3.29) uvedu pouze jeden příklad pro přechod a jeden příklad pro stav, s využitím koncentrovaných historických dat. Jako stav vyberu stav q_1 „napouštění nádrže 41“ a jako přechod vyberu $\alpha_{q1 \rightarrow q2}$, přechod ze stavu napouštění nádrže do stavu ukončení napouštění aditiv.

Pro dobu trvání stavu q_1 „napouštění nádrže“ vezmu změřené hodnoty z reálného procesu:

měření	1	2	3	4	5	6	7	8	9	10
doba trvání α_{q1}	598	605	600	595	605	598	606	600	605	602
Medián	600					602				

měření	11	12	13	14	15	16	17	18	19	20
doba trvání α_{q1}	595	602	600	598	599	600	603	598	603	597
Medián	599					600				

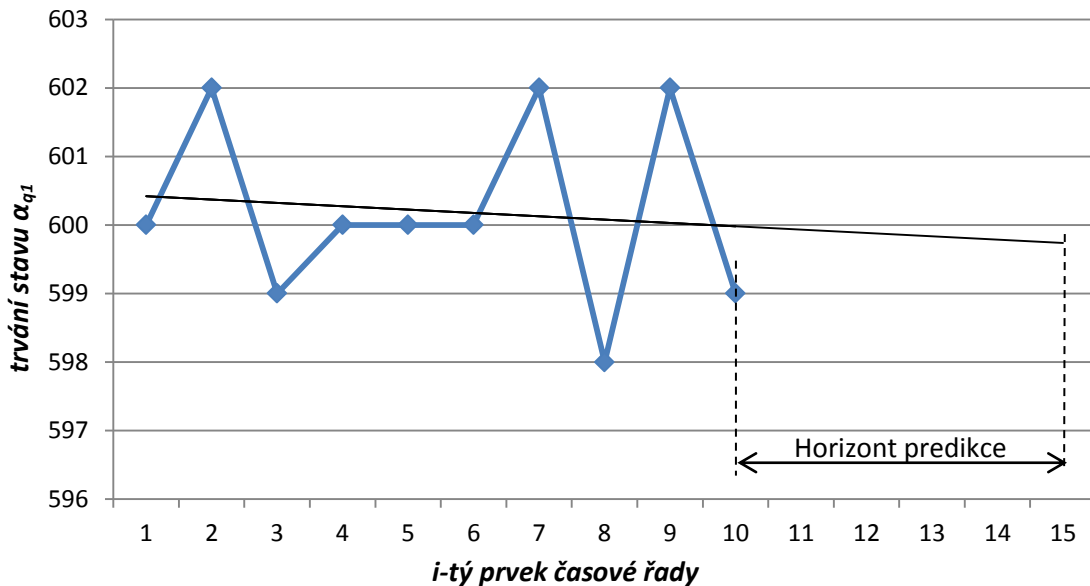
měření	21	22	23	24	25	26	27	28	29	30
doba trvání α_{q1}	600	599	601	600	598	600	602	598	602	599
Medián	600									

Obr. 31 Tabulka změřených hodnot doby trvání α_{q1}

Hodnoty časové řady pro stav q_1 jsou patrné z tabulky na Obr. 31, kdy pro měření 1 až 25 vezmu koncentrovaná „historická“ data (medián po pěti měřeních) a měření 26 až 30 budou poslední aktuální hodnoty potom

$$\alpha_{q1} = (600; 602; 599; 600; 600; 600; 602; 598; 602; 599)$$

Příslušná časová řada stavu α_{q1} včetně její linearizace je uvedena na Obr. 32.



Obr. 32 Časová řada stavu α_{q1} a její linearizace

Přechod $\alpha_{q1 \rightarrow q2}$ je přechod ze stavu napouštění nádrže do stavu ukončení napouštění. Napouštění je ukončeno po dosažení požadovaného množství aditiv v nádrži M41 a uzavření příslušných ventilů V31, V32 a V33 a dále po vypnutí čerpadel Č31, Č32 a Č33. V tomto příkladu jsem navrhl model jako společný pro všechny tři napouštěcí větve, tím vlastně sleduji celkovou dobu napouštění a rozhodující pro celkovou dobu plnění (trvání stavu $q1$) je dosažení požadovaného množství, uzavření dávkovacího ventilu a příslušného čerpadla u nejdéle se plnícího média. Jinak je samozřejmě možné realizovat model pro každou větev samostatně jako je tomu v Příkladu 1 uvedeném v kapitole 3.4.1.1 této práce. Stav $q1$ je ukončen povelům systému na odstavení posledního plnícího čerpadla a uzavření příslušného dávkovacího ventilu. Doba jejich přestavení je doba přechodu ze stavu $q1$ do stavu $q2$ tedy $\alpha_{q1 \rightarrow q2}$. Opět budu pracovat s aktuálními i historickými daty která budu koncentrovat. Hodnoty jsou uvedeny v tabulce na Obr. 33. Opět 1. až 25. hodnota jsou koncentrovány a následujících pět hodnot je aktuálních, $\alpha_{q1 \rightarrow q2}$ tedy bude následující.

$$\alpha_{q1 \rightarrow q2} = (32; 32; 32; 33; 32; 32; 31; 32; 33; 32)$$

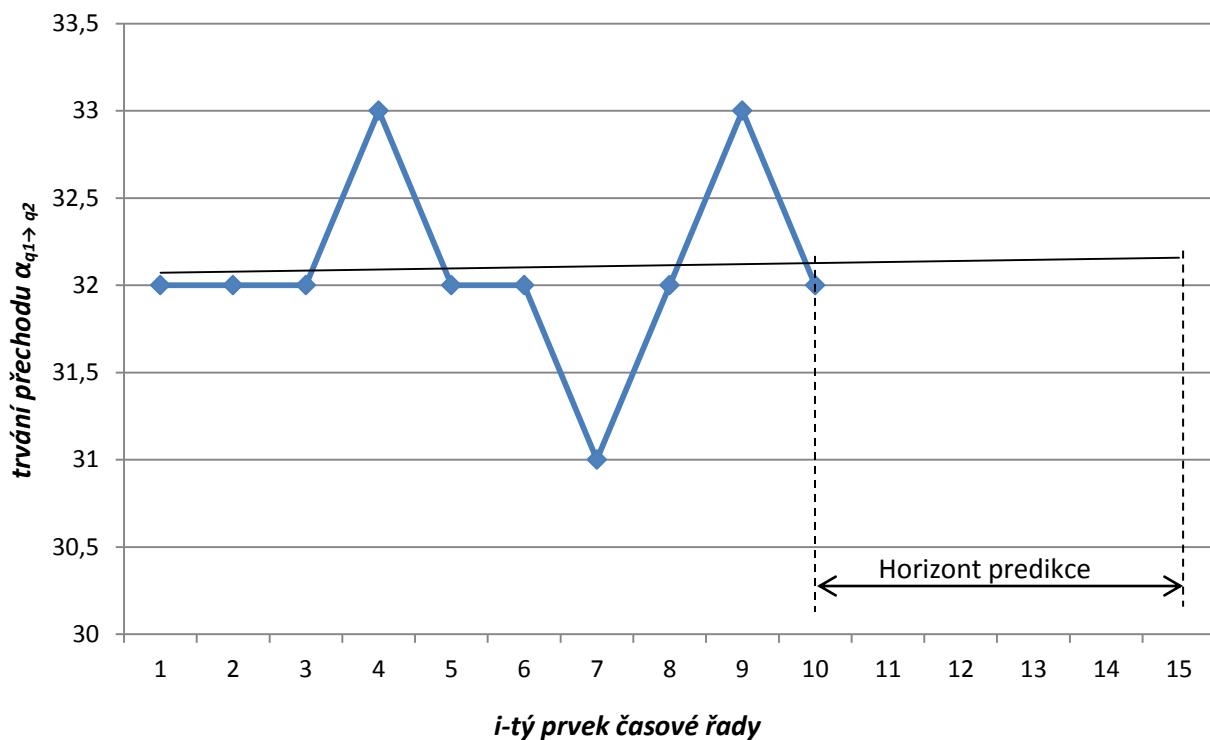
Příslušná časová řada přechodu $\alpha_{q1 \rightarrow q2}$ včetně její linearizace je uvedena na Obr. 34.

měření	1	2	3	4	5	6	7	8	9	10
doba $\alpha_{q1 \rightarrow q2}$	30	35	32	31	32	35	35	30	31	32
Medián	32					32				

měření	11	12	13	14	15	16	17	18	19	20
doba $\alpha_{q1 \rightarrow q2}$	31	30	32	32	35	35	33	37	30	29
Medián	32					33				

měření	21	22	23	24	25	26	27	28	29	30
doba $\alpha_{q1 \rightarrow q2}$	32	32	31	29	32	32	31	32	33	32
Medián	32									

Obr. 33 Tabulka změřených hodnot doby trvání $\alpha_{q1 \rightarrow q2}$



Obr. 34 Časová řada stavu $\alpha_{q1 \rightarrow q2}$ a její linearizace

Z výše uvedených grafických závislostí (*Obr. 22 až Obr. 30 a Obr. 32 a Obr. 34*) je patrné, jak se pravděpodobně bude chovat technologický proces v horizontu predikce. Na základě pravděpodobného budoucího chování lze diagnostikovat závady, nebo nestandardní chování technologického procesu. Podle toho, ve kterém diagnostikovaném uzlu dochází k problémům lze určit, kterého zařízení, nebo části procesu se tento problém týká, protože daný diagnostikovaný uzel je ovlivňován zařízeními a měřeními, která lze jasně určit. Např. pokud v naší vzorové úloze „Příklady technologického procesu 1-Výroba produktu“ podle odstavce 3.4.1.1, dochází ke zkracování doby trvání stavu q_4 , viz *Obr. 29* (v uvedené úloze se jedná o stáčení finálního produktu). Znamená to, že pokud je produkt stáčen rychleji než standardně, je pravděpodobné, že se zvyšuje průtok vypouštěcím ventilem, případně produkt uniká průběžně v době rekce, nebo došlo k posunutí nuly u měření hladiny v nádrži. Výsledkem je nutnost brzké kontroly technologického zařízení.

Znamená to tedy, že přechody a stavy musí být ztotožněny se zařízením, kterých se týkají a následně lze nadefinovat možné poruchy, včetně jejich příčin, které se týkají příslušného diagnostikovaného uzlu. Dále je zřejmé, že měření bude tím více vypovídající, čím déle bude prováděn sběr dat.

4. Závěr

Hlavním přínosem této práce je představení nového pohledu na moderní řízení, pohledu, že moderním řízením nejsou pouze vysoce sofistikované regulace, ale jsou zde i další ještě ne zcela využití možnosti řízení. Nejedná se vždy o úplně nové principy, ale jde i o využití stávajících možností řízení jiným než dosud využívaným způsobem. Práce měla za úkol představit prediktivní diagnostiku technologického procesu. Z možných prediktivních metod, které jsou k dispozici a které byly v práci popsány, jsem vybral predikci pomocí modelů. Predikce pomocí modelů totiž neřeší pouze predikování budoucí hodnoty určité veličiny, ale zaujímá komplexní pohled na celou soustavu, případně technologický proces, který je předobrazem modelu.

Po přijetí definice technologického procesu jako souboru stavů a událostí, které jsou známé, případně předpokládané a zejména skutečnosti, že jejich počet je konečný, jsem v práci navrhl prediktivní model pro diagnostiku, který vychází z teorie konečných automatů. Jako model technologického procesu jsem tedy v práci navrhl konečný automat pro diagnostiku. Konečný automat jako model byl vybrán proto, že technologický proces se ve své podstatě chová obdobně jako konečný automat, proto je možné principy konečného automatu s úspěchem využívat pro modelování technologického procesu. Konečný automat byl původně navržen pro účely v informatice a nikoliv pro řízení technologického procesu, nebo pro diagnostiku, v práci byl proto nově odvozen konečný automat pro diagnostiku a popsána jeho funkce. Základem tohoto automatu je deterministický konečný automat, který jsem pro diagnostické účely doplnil o maximální a minimální doby trvání stavů a přechodů, a zejména o časové řady pro tzv. diagnostikované uzly, jejichž vyhodnocením získáme informace o stavu technologického procesu a které tvoří podstatu predikce. Hlavní částí prediktivní diagnostiky je nadefinování prediktivních uzlů a shromažďování naměřených dat ve formě příslušných časových řad α ze kterých, některou z metod predikce, lze určit budoucí vývoj chování technologického procesu v příslušném prediktivním uzlu. Prediktivní uzel vždy náleží k určité části technologie a vývoj hodnot v příslušném uzlu tedy vždy odpovídá určité části technologického procesu.

5. Literatura

- [1] FERKL, L.: *Prediktivní regulace budov: matematika v praxi*. Stavebnictví 01/13, roč. VII, s. 42-46.
- [2] JAJČIŠIN, Š. JADLOVSKÁ, A.: *Adaptívne prediktívne riadenie na báze analytickej identifikácie riadeného systému*. Electrical Engineering and Informatics III, Proceeding of the Faculty of Electrical Engineering and Informatics of the Technical University of Košice, ISBN 978-80-553-0890-6 2012: 6 s.
- [3] GAJJAR, J. T. *PC-based diagnostic tool for predictive maintenance of cement mill equipment* Cement Industry Technical Conference, 1989, Record of Conference Papers, IEEE, 1989 : s. 215 -223
- [4] STEHLING, L ALANIS, J.: *Electric Heat Tracing Controls: Its Integration in Industrial Facilities* Industry Application Magazine, IEEE (Volume 18, Issue 2), 2012: s. 60 -69
- [5] HARTSHORN, G. LANZ, B. BROUSSARD, B.: *Medium Voltage Cable Predictive Diagnostic Technique*. IEEE PES PCIC Conference, September 19, 2007: s. 1-9
- [6] IEEE 400-2001 *Guide for Field Testing and Evaluation of the Insulation of Shield Power Cables*, Piscataway, NJ, IEEE, 2001: 40 s.
- [7] BATZEL, T.D. SWANSON, D.D. DEFENBAUGH, J.F.: *Predictive Diagnostics for the Main Field Winding and Rotating Rectifier Assembly in the Brushless Synchronous Generator*, SDEMPED 2003, USA, August, 2003: s. 24 -26
- [8] BYINGTON, C.S. GARGA, A.K.: *Data fusion for developing predictive diagnostics for electromechanical systems* In: HALL, D. LLINAS, J. , *Handbook of Data Fusion*, Boca Raton: CRC Press , 2001: s. 23-1 až 23-31
- [9] BAYES, T.: *An essay towards solving a problem in the doctrine of chances*, *Philosophical Trans. Royal Soc. London*, 53,1763: s. 370–418
- [10] JEONG, I-J. LEON, V.J. VILLALOBOS, J.R.: *Integrated decision-support system for diagnosis, maintenance planning, and scheduling of manufacturing systems*, *International Journal of Production Research*, Vol 45 No. 2, 15.1.2007: s 267-285
- [11] AJITH, A.: *Rule-based Expert Systems*, In *Handbook of Measuring System Design*, ed. SYDENHAM, P.H. TORN, R. John Wiley & Sons, Ltd. ISBN: 0-470-02143-8, 2005: s. 909 - 919

- [12] YE, N ZHAO, B.: *A Hybrid intelligent system for fault diagnosis of advanced manufacturing system*, Int. J. Prod. Res., 1998: s. 555-576
- [13] MOBLEY, R. K.: *An introduction to predictive maintenance*, Elsevier Science (USA), 2002: 437 s.
- [14] WESTINGHOUSE Electric Corp.: *Plant Maintenance with predictive diagnostics*. Původce vynálezu: PALUSAMY, S.S.; BAUMAN, D.A.; KOZLOVSKY, T.A.; BOND, C.H.B.; CRANFORD, E.L. III; BATT, T.J. Patent US 5311562 . United States Patent 10.05.1994
- [15] GTECH Corporation: *Predictive Diagnostics and Fault Management*. Původce vynálezu: GADHER, B.K . Patent US 8028196 B2 . United States Patent 27.9.2011
- [16] SCHNEIDER Automation: *Systeme de diagnostic predictif dans un automate programmable*. Původce vynálezu: ROUSSEAU, Robert; BONNARD, François; LAFUITE, Alain. Patent EP 1390819 European Patent Office 25.02.2014. Také zveřejněný jako *Diagnostic system in a programmable Automaton* Patent US 6937962 . United States Patent 30.8.2005.
- [17] CERRTI, U. GIACOBINI, M. LIADERT, P.: *Prediction of Binary Sequences by Evolving Finite state Machines*. 5th International Conference, Evolution Artificielle, France October 2001: s. 42-53
- [18] O'CONNOR, M.G.: *An unpredictability approach to finite state randomness*. Journal of Computer and System Science (37), 1988: s. 324-336
- [19] BROGLIO, A. LIADERT, P.: *Predictions with automata. symbolic dynamics and its applications*, Contemporary Mathematics 135, 1992: s. 111–124. Also in Proceedings AMS Conference in honor of R. L. Adler. New Haven CT - USA 1991.
- [20] RYABKO, B.Y.: *The Complexity and Effectiveness of Prediction Algorithms* Journal of Complexity 10, 1994: s. 281-295
- [21] ELMAN, J.L.: *Finding Structure in Time*. Cognitive science, 1990, 14.2: s. 179-211
- [22] CLEEREMANS, A SCHREIBER, D.S. Mc CLELLAND, J.L.: *Finite State Automata and Simple Recurrent Networks* Neural Computation 1, Massachusetts Institute of Technology 1989: s. 372-382
- [23] FEGWEI, S LIJUAN Y.: *Study on the Modeling Method of Software Process based on Timing and Parallel Automata* Proceedings of the 2013 2nd International Conference On Systems Engineering And Modeling (ICSEM-13) Published by Atlantis Press, Paris, France 2013 [doi:10.2991/icsem.2013.87](https://doi.org/10.2991/icsem.2013.87)
- [24] HURÁK, Z.: *Moderní metody řízení vs. PID regulátor*. Automa 2007, roč. 13, č.2: s.7 12.

- [25] ŠEBEK, M.: *Automatické řízení – nové trendy ve výzkumu a výuce*. Přednáška na semináři „Perspektivy automatizace“ Výstaviště Brno 20. 3. 2012 v rámci doprovodného programu veletrhu Amper 2012: 86 s.
- [26] HAVLENA, V. ŠTECHA, J.: *Moderní teorie řízení*, ČVUT Praha, 1999: 297 s.
- [27] CHALUPA, P.: *Prediktivní řízení nelineárního systému*. Univerzita Tomáše Bati ve Zlíně, 2006: 9 s.
- [28] POHL, T.: *Predikce v řízení a technické diagnostice*, Automatizace 2009, č. 10: s. 561-562
- [29] NYCE, CH.: *Predictive Analytics, White Paper*. American Institute for CPU, Insurance Institute of America 2007: 24 s.
- [30] TONISSEN, S.: *Automatické řízení – nové trendy ve výzkumu a výuce*. Řízení a údržba průmyslového podniku 2010, roč. III : s. 34-37
- [31] POHL, T.: *Metody predikce spojitéch veličin*, Automatizace 2009, č. 11: s. 656-657
- [32] CAMACHO, E.F. BORDONS, C.: *Model Predictive Control*. Springer-Verlag GmbH 2004: 405 s.
- [33] DOSTÁL, P. GAZDOŠ, F.: *Řízení technologických procesů*. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky 2006, Přístup z Internetu: <http://rtp.webzdarma.cz/>
- [34] Finite-state machine In *Wikipedie : otevřená encyklopedie* [online]. St. Petersburg (Florida) : Wikimedia Foundation, 2001- , strana naposledy edit. 2013-08-12. Anglická verze. Dostupný z WWW: < http://en.wikipedia.org/wiki/Finite-state_machine>.
- [35] CARROLL, J. LONG, D.: *Theory of Finite Automata with an introduction to formal language* Prentice-Hall, Englewood, USA 1989: 438 s.
- [36] XAVIER, S.P.E.: *Theory of Automata, Formal Languages and Computation* New Age International Ltd., Publishers 2005: 345 s.
- [37] MARTIN, J.C.: *Introduction to Languages and The Theory of Computation, fourth edition* McGraw-Hill, USA 2011: s. 45-77
- [38] LAWSON, M.V.: *Finite Automata*. CRC Press LLC 2004: 307 s.
- [39] LAWSON, M.V.: *Finite Automata*. School of Mathematical & Computer Sciences dostupný z <http://www.ma.hw.ac.uk/~markl/preprints/Lawson.pdf> : 27 s.
- [40] LAWSON, M.V.: *Finite Automata*. Heriot-Watt University, Edinburgh, November 4, 2009: 186 s
- [41] STRAUBIG, H. WEIL, P.: *An Introduction to Finite Automata and their Connection to Logic* September 21 2011 dostupný z <http://arxiv.org/pdf/1011.6491.pdf>

- [42] WRIGHT, D.R. *Finite State Machines*. CSC215 Class Notes. Prof. David R. Wright website, N. Carolina State Univ. Retrieved July 14, 2012: 28 s.
- [43] KARHUMAKI, J. *Automata and Formal Languages* 2005 dostupný z <http://www.math.utu.fi/en/home/karhumak/automata05.pdf> : 97 s.
- [44] LINDEN, R. SILVA, V.N.A.L.: *Finite Non-Deterministic Automata for Fault Diagnostic in Power System*. INFOCOMP, v.11 no 2, June 2012: s. 21 – 26
- [45] KOCUR, P.: *Úvod do teorie konečných automatů a formálních jazyků*. Západočeská univerzita v Plzni, 2001,. ISBN 80-7082-813-7: 104 s.
- [46] KOCUR, P.: *Úvod do teorie konečných automatů a formálních jazyků*. Západočeská univerzita v Plzni, 2005, e-book
Přístup z Internetu: <http://www.kvd.zcu.cz/cz/materialy/kafj/kafj1.html>
- [47] KOT, M. SAWA, Z.: *Minimalizace automatů (on-line)*. Ostrava: VŠB–TU Ostrava, 2007: 31 s. Přístup z Internetu: <http://www.cs.vsb.cz/kot/down/uti2007/uti-pr-04.pdf>
- [48] ROVAN, B. FORIŠEK, M.: *Formálne jazyky a automaty*. Univerzita Komenského Bratislava, Katedra Informatiky, skripta březen 2011: 114 s.
- [49] DOSTÁL, H.: *Teorie konečných automatů, regulárních gramatik, jazyků a výrazů*. Internetové podpůrné výukové prostředí k předmětu teoretická informatika, Univerzita Hradec Králové <http://iris.uhk.cz/tein/index.html>
- [50] JANČAR, P.: *Studijní opora k předmětu Teoretická informatika*. 25. září 200: 96 s.
Přístup z Internetu: http://www.cs.vsb.cz/jancar/TJAA/tjaa_2p.pdf