

České vysoké učení technické v Praze
Fakulta elektrotechnická
Obor: Technická kybernetika

Název diplomové práce:

”Detekce poruchových stavů dynamických systémů”

Vypracoval: Tomáš Fibír
Vedoucí diplomové práce: Ing. Daniel Pachner

PROHLÁŠENÍ

Prohlašuji, že jsem svou diplomovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

Nemám závažný důvod proti užití tohoto školního díla ve smyslu § 60 Zákona č.121/2000 Sb., o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Praze, dne

Podpis

PODĚKOVÁNÍ

Děkuji panu Ing. Danielu Pachnerovi za pomoc a konzultace při zpracování diplomové práce. Dále děkuji společnosti PTC Honeywell za poskytnuté informace a podmínky pro vytvoření této práce.

Abstrakt

Tématem této diplomové práce je detekce poruchových stavů lineárního dynamického systému, u kterého jsme schopni sestavit jeho analytický model. V úvodní kapitole jsou popsány některé obecné aspekty detekce poruch, jako např. obecná formulace problému, členění různých přístupů, apod. Samotná práce se pak zabývá popisem dvou vybraných algoritmů pro detekci poruch. Je to především výpočet minimální normy lineárních poruchových signálů pro deterministický systém, a poté výpočet pravděpodobnosti různých kovariančních matic (reprezentujících různé poruchy) náhodných poruchových signálů pro stochastický systém. Důraz je kláden na odvození rekurzivnosti obou algoritmů a dále na jejich numerickou stabilitu a efektivnost. Funkčnost obou algoritmů je ověřena na jednoduchém systému. Pomocí algoritmu pro stochastický systém je řešen reálný problém detekce poruch analyzátorů spalin při hoření uhlí v teplárně Otrokovice. V závěru jsou diskutovány dosažené výsledky.

Abstract

The topic of this diploma work is detection of fault states of linear dynamic system, which is described by known analytical model. In the first chapter there are described some general aspects of fault detection, e.g. general formulation of the problem, itemization of various concepts, etc. The main part of this work describes two algorithms. At first it is the computation of minimal quadratic norm of linear fault signals for deterministic systems, at second it is the computation of probabilities of different covariance matrixes (they represent different faults) of random fault signals for stochastic systems. The emphasis is given on the derivation of recurrent versions of both algorithms and also on their numerical stability and efficiency. The functionality of both algorithms is validated on one simple system. One real practical problem of fault detection of the flue gas analyzers in the process of coal combustion in the boiler plant Otrokovice is solved with a help of the algorithm for stochastic systems. The results are discussed in the final.

Obsah

Úvod	3
1 Detekce poruch-obecné poznatky	5
1.1 Fáze detekce poruch	5
1.2 Obecná definice poruchy	5
1.3 Přístupy k detekci poruch	5
1.3.1 Paralelní modely	6
1.3.2 Generování reziduí	6
1.4 Formulace problému	7
1.5 Lineární systém s lineárními poruchami	8
1.6 Minimální norma poruchových signálů	9
1.7 Řešení	10
1.7.1 Řešení pro $Q=I$	10
1.7.2 Řešení pro obecnou matici Q	11
2 Deterministický přístup - demonstrace na příkladě	12
2.1 Konkrétní systém	12
2.2 Lineární omezení	13
2.3 Minimalizace kvadratické formy	14
3 Rekurzivní algoritmus minimalizace	17
3.1 Motivace	17
3.2 Kvazidiagonální tvar lineárních omezení	17
3.3 Rekurzivní algoritmus	20
3.3.1 Pomocné pojmy	20
3.3.2 Úprava minimalizované kvadratické formy	20
3.3.3 Minimální hodnota kvadratické formy	21
3.3.4 Zahrnutí omezení do matice F	22
3.3.5 Úprava na zobecněný choleskyho faktor	26
3.3.6 Vlastní minimalizace	27
3.3.7 Závěrečná úprava	29

3.3.8	Shrnutí	29
3.4	Příklad	30
4	Stochastický přístup	36
4.1	Stochastický model s poruchami	36
4.2	Poruchový signál jako náhodný vektor	37
4.3	Význam kovarianční matice pro detekci	37
4.4	Výpočet pravděpodobnosti kovarianční matice	38
4.5	Rekurzivní výpočet pravděpodobnosti kovarianční matice	39
4.5.1	Motivace	39
4.5.2	Formální úpravy	39
4.5.3	Integrace hustoty pravděpodobnosti přes lineární podprostor	40
4.5.4	Rekurzivní integrace	42
4.6	Příklad	44
5	Aplikace na reálném případě z praxe - Řízení spalování uhlí v teplárně Otrokovice	48
5.1	Úvod	48
5.2	Řídící proces	48
5.3	Úkol detekce poruch	49
5.4	Struktura systému	49
5.5	Parametry modelu	53
5.6	Vlastní detekce poruch	56
Závěr		65
Literatura		67

Úvod

Tématem zpracovávaným v této diplomové práci je detekce poruchových stavů daného systému, konkrétně lineárního (spojitého nebo diskrétního) systému, u kterého jsme schopni sestavit jeho analytický model, například ve formě diferenciální (diferenční) rovnice.

Neustálý rozvoj automatizovaného řízení a jeho pronikání do stále rozsáhlejších a složitějších průmyslových procesů, jakou jsou např. chemické procesy či procesy spojené s leteckým průmyslem, vede k potřebě zabezpečit celý řídící systém proti různým očekávatelným poruchám. Aby byl celý řídící systém skutečně plně automatizovaný a tím pádem nezávislý na lidských zásazích z venčí, je nasnadě požadovat, aby výše zmíněné zabezpečení proti poruchám bylo rovněž automatické a tím bylo možno jej do řídícího systému plně začlenit. Prvním a základním krokem k navržení takového řídícího systému odolného vůči poruchám je automatická detekce těchto poruch, což, jak již bylo řečeno, je náplní této práce.

V úvodní kapitole jsou uvedeny některé obecné poznatky o detekci jako takové. Jsou zde popsány dva základní přístupy, podle nichž lze jednotlivé algoritmy detekce dělit. Dále je zde popsáno rozdělení vlastní detekce na jednotlivé fáze a také obecnější formulace celého problému detekce, což umožňuje začlenění tohoto problému do širších souvislostí z oblasti statistického rozhodování. Součástí této kapitoly je rovněž popis a principiální řešení přístupu, který tvoří jádro celé práce a který je v dalších kapitolách rozváděn. Jedná se o výpočet minimální normy poruchových signálů působících na daný systém.

V dalších dvou kapitolách, které tvoří jakousi první část celé práce, je nejprve demonstrován výpočet minimální normy na konkrétním příkladě, což má demonstrovat význam minimální normy pro detekci. A poté je podrobně popsán celý algoritmus pro výpočet minimální normy. Hlavním téžistěm této části je převedení principiálního řešení na rekurzivní algoritmus, který je již možno použít pro praktický výpočet, při kterém se předpokládá neustálý příspun nových dat získaných měřením na daném systému. Vedle rekurzivnosti algoritmu, jakožto základní a nutné vlastnosti, je kladen důraz rovněž na numerickou stabilitu a efektivnost celého algoritmu. Celý algoritmus je pak

demonstrován opět na příkladě.

V druhé části diplomové práce je přístup založený na výpočtu minimální normy poruchových signálů rozšířen na stochastické systémy, což jsou systémy, u kterých již nelze neurčitosti ve vnitřní struktuře zanedbat. Přestože v tomto případě jsou namísto minimální normy poruchových signálů počítány pravděpodobnosti různých kovariančních matic těchto signálů a tento přístup tedy vychází ideově z jiných principů a používá jiných pojmu, jsou oba přístupy z hlediska poskytovaných výstupů srovnatelné, což je také na začátku této části ukázáno.

Závěrečná praktická část zaměřuje pozornost na aplikaci popsaného algoritmu na konkrétním problému detekce poruch analyzátorů spalin při spalování uhlí v teplárně Otrokovice. Je věnována nejen samotné aplikaci algoritmu automatické detekce, ale i problému sestavení modelu vhodného právě pro tuto detekci.

Kapitola 1

Detekce poruch-obecné poznatky

1.1 Fáze detekce poruch

Celou detekci poruch lze rozdělit do tří fází (viz [3]). První, kterou se budeme v této práci zabývat především, je vlastní detekce poruch (fault detection), tedy detekování času, kdy začala působit porucha. Druhou je izolace poruch (fault isolation), což je lokalizace (klasifikace) poruchy. Je to v podstatě logický rozhodovací proces založený na statistickém rozhodování. A třetí fází je analýza poruch (fault analysis), tedy určení typu, velikosti a zdroje poruchy (za účelem stanovení patřičných opatření). Tato fáze je ve většině případů nejednoznačná a závisí na počtu měřených signálů v daném procesu, což bývá v praxi limitující faktor.

1.2 Obecná definice poruchy

Porucha je náhodný neměřitelný proces. Kdybychom poruchu mohli měřit, neměla by úloha detekce smysl. O poruše se můžeme dozvědět pouze na základě dat, které pozorujeme (měříme) na systému.

1.3 Přístupy k detekci poruch

V zásadě můžeme rozlišovat dva možné statistické přístupy k detekci poruch. Jeden přístup používá explicitní model změny parametrů, zatímco druhý nikoliv.

1.3.1 Paralelní modely

Do skupiny používající explicitní model změny parametrů řadíme například metody označované jako paralelní modely, u nichž můžeme rozlišovat dva možné případy

- paralelní modely - porucha působí na systém od začátku, tzn. od okamžiku, kdy jsme začali získávat data, a naším úkolem je na základě porovnání výstupů jednotlivých modelů s výstupem reálné soustavy určit, která z poruch je v daném okamžiku aktivní. V tomto případě je základním problémem, jak se změny parametrů soustavy podléhající působení poruchy promítou na výstup soustavy, jinými slovy, jak zvolit práh normálního (bezporuchového) chování soustavy.
- paralelní interagující modely - v tomto případě uvažujeme poruchy působící od libovolného časového okamžiku po začátku měření až po nynější okamžik. Cílem je určit, která z poruch je nyní aktivní, popřípadě od kdy.

1.3.2 Generování reziduí

Druhý přístup nepoužívá explicitní model změny parametrů. Je založen na generování reziduí (příznaků), což se děje porovnáním dat z procesu a odpovídajících referenčních hodnot získaných za bezporuchového stavu. Generátor reziduí je tedy vlastně dynamický systém (filtr), který je řízen vstupy a výstupy procesu. Základní vlastností reziduí je, že jsou nereditelné počátečními podmínkami soustavy a poruchami, jejichž vliv není kritický na chování celého procesu (např. šum (nepřesnost) měření, nepřesnost modelu apod.). Necitlivost na takové typy poruch je někdy označována jako robustnost a je tedy základním požadavkem na detekci poruch. Při splnění těchto podmínek nám potom nulovost či nenulovost reziduí rozhoduje o přítomnosti či nepřítomnosti poruchy.

Problémem většiny algoritmů (např. CUSUM v [1]) založených na sekvenčním rozhodovacím procesu (který vlastně generuje rezidua) je používaný předpoklad, že zpracováváme posloupnost nezávislých stejně rozdělených veličin (i.i.d. posloupnost). Tento předpoklad ale při použití k detekci poruch dynamických systémů není splněn, neboť data získaná z dynamických systémů jsou závislá. Proto jsme ke generování reziduí přistoupili jinak a jako rezidua používáme dolní odhad kvadratické normy (jinak také energie) poruchových signálů.

1.4 Formulace problému

Proved'me nyní formálnější formulaci našeho problému. Nejprve je dobré zdůraznit, že v našem případě detekujeme pouze modelované poruchy příslušné danému modelu. Nikoliv poruchy jakékoliv. Typicky to mohou být v zásadě tři typy poruch: poruchy akčních členů, poruchy složek procesu a poruchy čidel. Všechny tyto poruchy mohou být obecně popsány jako neznámé (na rozdíl od známých vstupů do soustavy) vstupní signály (budeme je nadále značit z).

Řekněme tedy, že máme model $M(\theta)$, $\theta \in \Theta$, který generuje na výstupy data $y \in Y$. Parametr θ je množina proměnných, která popisuje nějaký stav dané soustavy. Zařízení, které provádí detekci poruch (označme jej F), má na svém vstupu pozorovaná data y a na svém výstupu alarmový signál $a \in A$, kde A je množina všech možných alarmů. Provádí tedy zobrazení $z \rightarrow a$.

Nyní můžeme detekci poruch definovat jako úlohu, která pro daný model $M(\theta)$ hledá takové a , že funkce $L(a, \theta)$ nabývá minimální hodnoty. Funkce L je ztrátová funkce, která ke každé dvojici (a, θ) přiřadí nezápornou hodnotu, kterou můžeme interpretovat jako ztrátu, kterou utrpíme, budeme-li při dané hodnotě parametrů θ generovat alarm a .

Takto definovaná úloha je optimalizační úloha. Vzhledem k tomu, že neznáme přesnou hodnotu parametru θ , ale můžeme vyjádřit hustotu pravděpodobnosti neznámých parametrů podmíněnou pozorovanými daty, lze určit optimální rozhodnutí (hodnotu a) jako rozhodnutí, které minimalizuje střední hodnotu $L(a)$ (Bayesovské rozhodování)

$$R(a) = \varepsilon\{L(a)\} = \int_{\Theta} L(a, \theta) f(\theta|y) dp = \int_{\Theta} L(a, \theta) f(y|\theta) g(\theta) dp, \quad (1.1)$$

kde $g(\theta)$ je apriorní hustota pravděpodobnosti parametrů θ .

Nyní si přesně definujeme, co budeme v našem případě označovat za poruchu. Na námi uvažovaný systém (s neznámými poruchovými signály z) začala v intervalu $<0, T>$ působit k -tá porucha, právě když

$$\left(p(T) - p_k(T) \right)^T Q_k \left(p(T) - p_k(T) \right) \geq \epsilon > 0 \text{ a píšeme } p(T) \in Z_k,$$

kde $p(T) = \{z(0), \dots, z(T)\}$ a Q_k , resp. p_k je jádro, resp. střed kvadratické formy.

O příslušnosti do množiny poruchových signálů tedy rozhoduje číselná charakteristika poruchy, konkrétně její kvadratická norma, která se dá interpretovat jako energie daného poruchového signálu.

Pro správnou (skutečnou) hodnotu alarmu γ_k potom platí

$$\gamma_k = \begin{cases} \nearrow 1 & \Leftrightarrow p(T) \in Z_k \\ \searrow 0 & \text{jinak} \end{cases}$$

Úloha definovaná vztahem 1.1 je tedy vlastně problém odhadu alarmu γ_k (tento odhad jsme označili a) tak, že minimalizujeme ztrátu danou ztrátovou funkcí L . Pro nás problém budeme uvažovat asymetrickou ztrátovou funkci

$$L(a, \gamma) = \begin{array}{cc} & \begin{matrix} \gamma = 0 & \gamma = 1 \end{matrix} \\ \begin{matrix} a = 0 \\ a = 1 \end{matrix} & \left(\begin{matrix} 0 & k \\ \infty & 0 \end{matrix} \right) \end{array}$$

Takováto ztrátová funkce approximuje situaci, kdy cena za falešný alarm je mnohem větší než cena za neohlášený alarm. Pro vlastní minimalizaci takovéto ztrátové funkce to znamená, že pouze zjišťujeme, zda je nenulová pravděpodobnost, že $\gamma_k = 0$. Na přesné velikosti této pravděpodobnosti nezáleží.

1.5 Lineární systém s lineárními poruchami

Jak již bylo řečeno výše, budeme řešit problém detekce poruch na třídě lineárních systémů. Proveďme nejprve definici stavového modelu lineárního (diskrétního) systému.

Máme stavový prostor $\mathcal{X} = \mathcal{R}^{n(x)}$, jehož jednotlivé elementy reprezentují jednotlivé stavy systému. Systém je dále buzen vektorem vstupů $u(t) \in \mathcal{U} = \mathcal{R}^{n(u)}$ a na výstupu produkuje data $y(t) \in \mathcal{Y} = \mathcal{R}^{n(y)}$. Pro lineární systém platí, že budoucí hodnoty stavů a současné hodnoty výstupů jsou lineární kombinací současných hodnot stavů a současných hodnot vstupů. To zapisujeme

$$\begin{aligned} x(t+1) &= \mathbf{A}x(t) + \mathbf{B}u(t) \\ y(t) &= \mathbf{C}x(t) + \mathbf{D}u(t), \end{aligned}$$

kde matice $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ jsou známé parametry a čas t nabývá hodnot z oboru přirozených čísel.

Existují i jiné způsoby popisu lineárního systému, ale ze stavového modelu je nejlépe patrná vnitřní struktura modelu, což je při uvažování poruch,

které do této vnitřní struktury přímo zasahují, velmi vhodné. Nyní definujme lineární systém rozšířený o lineární poruchové signály

$$\begin{aligned} x(t+1) &= \mathbf{A}x(t) + \mathbf{B}u(t) + \mathbf{F}z(t) \\ y(t) &= \mathbf{C}x(t) + \mathbf{D}u(t) + \mathbf{G}z(t), \end{aligned} \quad (1.2)$$

kde z je neměřitelný vektorový signál.

Druhá možná definice lineárního systému s lineárními poruchami je systém bez neměřitelného signálu, ale s časově proměnnými (nepredikovatelně) stavovými maticemi

$$\begin{aligned} x(t+1) &= (\mathbf{A} + \mathbf{d}\mathbf{A}(t))x(t) + (\mathbf{B} + \mathbf{d}\mathbf{B}(t))u(t) \\ y(t) &= (\mathbf{C} + \mathbf{d}\mathbf{C}(t))x(t) + (\mathbf{D} + \mathbf{d}\mathbf{D}(t))u(t). \end{aligned}$$

Je zřejmé, že oba modely jsou ekvivalentní, platí-li

$$\begin{aligned} \mathbf{F} &= [\mathbf{I} \quad \mathbf{O}] \\ \mathbf{G} &= [\mathbf{O} \quad \mathbf{I}] \\ z(t) &= \begin{bmatrix} \mathbf{d}\mathbf{A}(t)x(t) + \mathbf{d}\mathbf{B}(t)u(t) \\ \mathbf{d}\mathbf{C}(t)x(t) + \mathbf{d}\mathbf{D}(t)u(t) \end{bmatrix}. \end{aligned}$$

1.6 Minimální norma poruchových signálů

Budeme řešit problém, jak najít minimální hodnotu kvadratické normy neměřitelného poruchového signálu z , za podmínky splnění stavových rovnic našeho modelu pozorovanými daty. Budeme tedy řešit kvadratický optimalizační problém za lineárních omezení.

Nejprve si zavedeme označení pro soubor daných vektorů od času t_1 do času t_2 jako sloupcový vektor

$$x_{t_2}^{t_1} = \begin{cases} / & x(t_1) \quad \text{pro } t_1 = t_2 \\ \backslash & \begin{pmatrix} x_{t_2-1}^{t_1} \\ x(t_2) \end{pmatrix} \quad \text{jinak} \end{cases}$$

A nyní můžeme definovat naši optimalizační úlohu: Za předpokladu, že měříme vstupy a výstupy systému definovaném v (1.2), najděme minimální hodnotu kvadratické formy $(z_T^0)^T Q(z_T^0)$ (tuto hodnotu označme jako $c(T)$) při splnění soustav lineárních rovnic $Hz_T^0 = h$, tedy

$$\min_{Hz_T^0=h} (z_T^0)^T Q(z_T^0), \text{ kde } H = \begin{pmatrix} G & O & O & \cdots & O \\ CF & G & O & \cdots & O \\ CAF & CF & G & \cdots & O \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^T F & CA^{T-1} F & CA^{T-2} F & \cdots & G \end{pmatrix},$$

$$h = y_T^0 - \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^T \end{pmatrix} x(0) - \begin{pmatrix} D & O & O & \cdots & O \\ CB & D & O & \cdots & O \\ CAB & CB & D & \cdots & O \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^T B & CA^{T-1} B & CA^{T-2} B & \cdots & D \end{pmatrix} u_T^0$$

Tvar matice H a vektoru h jsme dostali rekurzivním dosazováním vztahů v (1.2).

1.7 Řešení

1.7.1 Řešení pro Q=I

Hodnotu $c(T)$ (tedy minimální hodnotu $z^T Q z$) nejprve určíme pro $Q = I$. Všechny vektory z splňující omezení ve tvaru $H z = h$, můžeme zapsat

$$z = H^- h + H_{ker} s, \text{ kde}$$

H^- je pseudoinverzní matice matice H , tedy platí $HH^-H = H$,

H_{ker} je ortogonální báze jádra zobrazení definované maticí H

s je libovolný vektor příslušné dimenze

Uvědomíme-li si, že matice H^- je tvořena vektory tvořící ortogonální bázi prostoru, který je komplementární k jádru zobrazení H , dostaneme $H_{ker}^T H^- = O$. S využitím tohoto poznatku můžeme upravit vztah pro kvadratickou formu $z^T Q z$

$$\begin{aligned} z^T Q z &= z^T z = (h^T H^{-T} + s^T H_{ker}^T)(H^- h + H_{ker} s) = \\ &= h^T H^{-T} H^- h + h^T H^{-T} H_{ker} s + s^T H_{ker}^T H^- h + s^T H_{ker}^T H_{ker} s = \\ &= \|H^- h\|^2 + \|H_{ker} s\|^2 \end{aligned}$$

A pro hodnotu $c(T)$ potom platí

$$c(T) = \min_s \left(\|H^- h\|^2 + \|H_{ker} s\|^2 \right) = h^T H^{-T} H^- h$$

1.7.2 Řešení pro obecnou matici \mathbf{Q}

Je-li $Q \neq I$, pak provedeme následující substituce

$$F \rightarrow F' = FQ_c^{-1}, \quad G \rightarrow G' = GQ_c^{-1}, \quad z \rightarrow z' = Q_c z,$$

kde Q_c je choleskyho faktor matice Q ($Q = Q_c^T Q_c$).

Je zřejmé, že těmito substitucemi jsme náš původní systém nezměnili, neboť platí

$$F' z' = Fz, \quad \text{resp. } G' z' = Gz.$$

A po dosazení do kvadratické formy $z^T Q z$ dostaneme tvar, pro který jsme řešení odvodili v předchozí části. Neboť

$$z^T Q z = z^T Q_c^T Q_c z = (Q_c z)^T (Q_c z) = z'^T z'.$$

Kapitola 2

Deterministický přístup - demonstrace na příkladě

Počítání minimální normy poruchového signálu z rozšířeného lineárního systému, který jsme definovali v kapitole 1.5, budeme označovat jako deterministický přístup k detekci poruch. V kapitole 4 se ještě seznámíme se stochastickým přístupem.

2.1 Konkrétní systém

Ilustrujme deterministický přístup na jednoduchém příkladě. Pro grafickou názornost budeme uvažovat systém s jedním vstupem a jedním výstupem (SISO) a dvěma poruchovými signály z . První bude reprezentovat chybu akčního členu, což znamená, že na soustavu nebude působit pouze námi měřený vstupní signál, ale bude ovlivněn také tímto prvním poruchovým signálem. Druhý poruchový signál bude reprezentovat aditivní poruchu senzoru měřené veličiny (tedy výstupu). Konkrétně tedy stavové rovnice našeho lineárního systému s lineárními poruchami mohou vypadat takto:

$$x(k+1) = \begin{pmatrix} 0 & 1 \\ 0 & 0.5 \end{pmatrix} x(k) + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u(k) + \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} z(k) \quad (2.1)$$

$$y(k) = (-1 \ 1) x(k) + (0 \ 1) z(k) \quad (2.2)$$

a matice A, B, C, D, F, G jsou tedy

$$\begin{aligned} A &= \begin{pmatrix} 0 & 1 \\ 0 & 0.5 \end{pmatrix} & B &= \begin{pmatrix} 0 \\ 1 \end{pmatrix} & F &= \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \\ C &= (-1 \ 1) & D &= (0) & G &= (0 \ 1) \end{aligned}$$

2.2 Lineární omezení

Jelikož máme dva poruchové signály, tak množina všech možných hodnot těchto poruchových signálů v daném časovém okamžiku tvoří rovinu. Měřením dat (vstupu a výstupu) dostáváme omezení na hodnoty těchto poruchových signálů, a to ve formě lineárních rovnic. Těchto rovnic je právě tolik, kolik měříme výstupů. Řešení těchto rovnic, tedy množina všech hodnot poruchových signálů, jejichž působení na daný systém by generovalo pozorované výstupy, tvoří lineární podprostor původního prostoru všech možných hodnot poruchových signálů. V našem jednoduchém případě bude dimenze tohoto podprostoru po naměření první dvojice vstupu a výstupu o jednu menší než původní prostor (což byla rovina). Bude to tedy přímka. Její rovnice bude :

$$Gz(0) = y(0) - Cx(0) - Du(0) \quad (2.3)$$

Je vidět, že směr této přímky je pevně dán stavovými maticemi rozšířeného modelu systému, neboť v matici definující tvar lineární formy (tzn. lineární omezení) G se nevyskytuje měřená data. Naopak data se vyskytují pouze na pravé straně, která reprezentuje hodnotu této lineární formy. V našem případě je tato hodnota reprezentována posunutím přímky od počátku.

Zamysleme se nyní, jak je to s nulovostí či nenulovostí této vzdálenosti ve vztahu k nulovosti či nenulovosti poruchových signálů. Tyto pojmy totiž nejsou ekvivalentní, jak by se možná mohlo na první pohled zdát.

Například z faktu, že na systém působí námi modelovaný poruchový signál z , nevyplývá, že hodnota lineární formy (2.3) je nenulová (neboli, že v našem případě přímka neprochází počátkem). K takovému stavu může dojít v zásadě ve dvou případech. Jedenak, není-li ta část dynamiky systému, která je poruchou ovlivněna, daty dostatečně buzena, a jednak v případě, kdy jiné neměřitelné signály, které ovšem nemodelujeme jako poruchy (například šum meření), kompenzují vliv právě onoho aktivního poruchového signálu. První případ mužeme jednoduše demonstrovat na našem příkladě, kdy díky neexistenci přímé vazby mezi vstupem a výstupem, není možné, aby se v prvním kroku měření detekovala porucha akčního členu, neboť tato porucha ovlivňuje chování systému právě prostřednictvím vstupu. Jiným triviálním příkladem by byl případ, kdy vstup soustavy by byl po určitou dobu nulový.

Rovněž nepůsobí-li na systém žádný z námi modelovaných poruch, nemusí být hodnota lineární formy nulová. To mohou způsobit další neurčitosti, které se v praxi samozřejmě vyskytují (například nepřesnost modelu). Z toho okamžitě vyplývá, že nulovost lineární formy nám nic určitého neříká o existenci či neexistenci poruchy.

Naopak, je-li hodnota lineární formy nenulová, mužeme říct, že na systém

působí porucha. Druhou otázkou zbývá, zda se nám podaří blíže specifikovat, o jakou poruchu přesně jde.

2.3 Minimalizace kvadratické formy

V předchozí části jsme ilustrovali, jak vypadá množina poruchových signálů, které splňují omezení dané naměřenými daty. Nyní budeme ilustrovat nalezení takového bodu (vektoru), který leží na naší přímce a zároveň minimalizuje kvadratickou formu $z^T Q z$, kde matice Q je jádro kvadratické formy. Matice Q je pozitivně definitní, takže každému nenulovému vektoru z přiřazuje kladné číslo. Fyzikální význam této hodnoty kvadratické formy je energie poruchového signálu z , neboť je to norma signálu (vektoru) $Q_c z$, kde Q_c je choleskyho faktor matice Q . Tedy

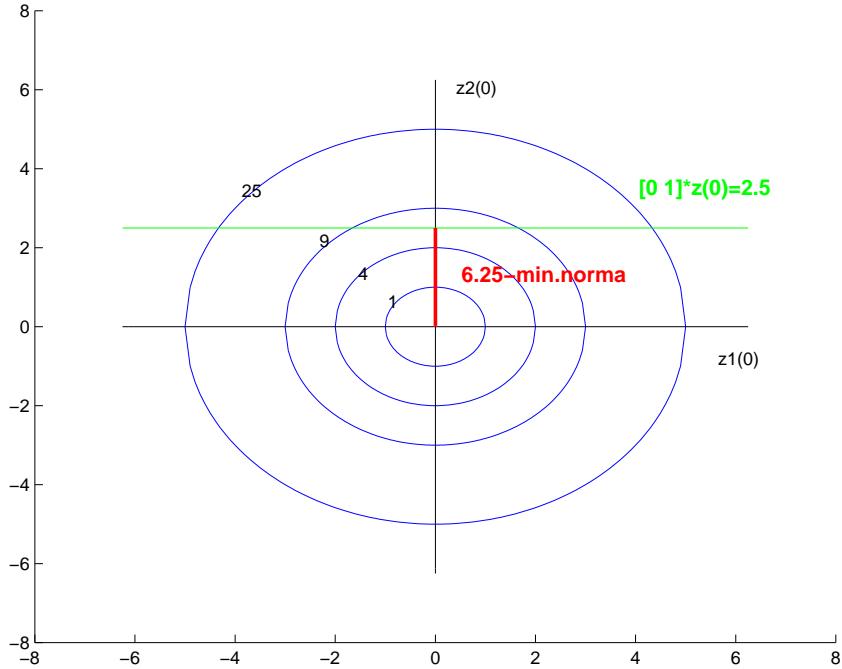
$$Q = Q_c^T Q_c, \quad z^T Q z = z^T Q_c^T Q_c z = \|Q_c z\|^2.$$

Proto je matice Q rovnež symetrická, neboť energie signálu z a $-z$ je stejná. Změnou koeficientů matice Q měníme citlivost hodnoty kvadratické formy na jednotlivých souřadnicích vektoru z . Při detekci poruch používáme výhradně diagonální matice Q a měníme tedy pouze koeficienty na diagonále, což nám v některých případech může pomoci při určení, o jakou poruchu se jedná (poté, co jsme detekovali, že k nejaké poruše došlo). Ilustrujme to na našem příkladě.

Pro vlastní detekci, zda vůbec došlo k nějaké poruše, je logické použít kvadratickou formu, která nerozlišuje mezi jednotlivými složkami signálu z a všechny váží stejnými koeficienty. Proto použijeme nejprve jednotkovou matici

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \text{ a } z^T Q z = z_1^2 + z_2^2.$$

Množiny vektorů z se stejnou hodnotou této kvadratické formy tvoří soustředné kružnice, pričemž tato hodnota roste směrem od počátku ($z = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$). Z teorie vázaných extrémů je známo, že pro bod minimalizující nějakou funkci za omezujících podmínek platí, že v tomto bodě je normálový vektor kriteriální funkce rovnobežný s normálovým vektorem omezení. Jelikož v našem případě je množina bodů splňujících omezení přímka, hledáme vlastně bod, ve kterém je tato přímka tečnou ke křivce spojující body o stejné hodnotě kvadratické formy. Pro jednotkovou matici Q je hodnota takové kvadratické formy v každém bodě zároveň kvadrátem vzdálenosti tohoto bodu od počátku (přesněji řečeno středu kvadratické formy). Proto můžeme námi hledané minimum najít jako průsečík přímky procházející počátkem, která



Obrázek 2.1: Minimální norma pro $Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$

je kolmá na přímku s přípustnými body, a této přímky (obr2.1). Vidíme, že jsme získali bod s nenulovou normou, a proto můžeme prohlásit, že jsme detekovali poruchu.

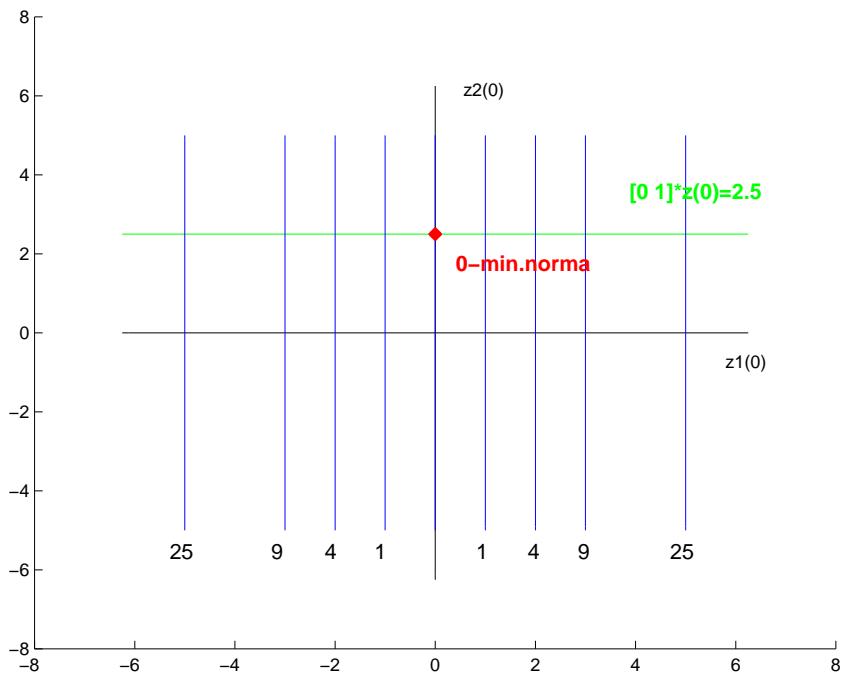
Nyní bychom chtěli zjistit, která porucha (popř. které poruchy) z námi modelovaných je aktivní. Proto místo jednotkové matice Q budeme uvažovat matici, která zařídí, aby velikost naší kvadratické formy nebyla ovlivněna vždy po řadě jedním z poruchových signálů. Nejprve tedy potlačíme například vliv druhé složky vektoru z . Matice Q proto bude mít tvar

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \text{ a } z^T Q z = z_1^2.$$

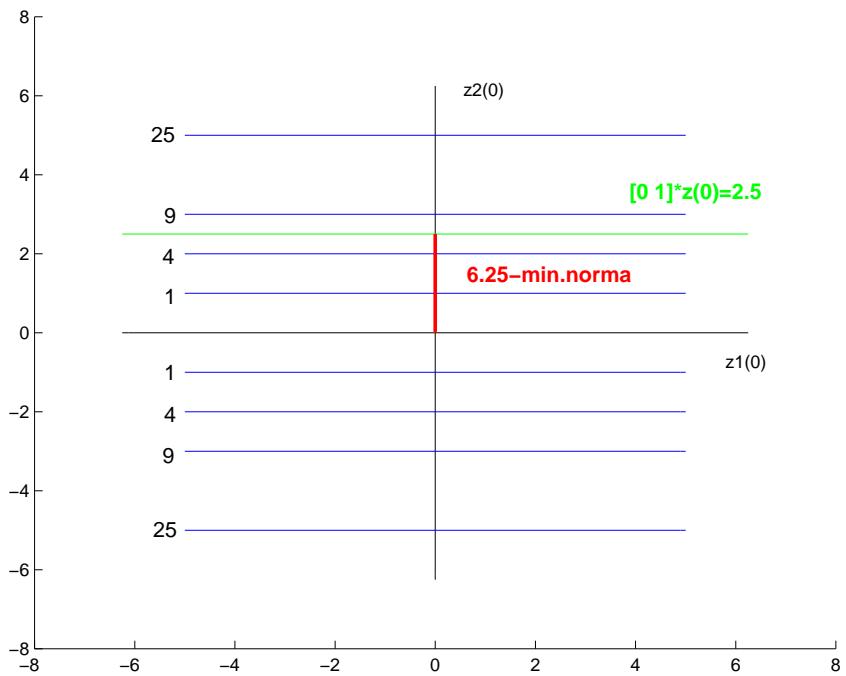
Množiny bodů o stejně hodnotě této formy jsou zřejmě přímky rovnoběžné s osou z_2 . Z obr2.2 je patrné, že tentokrát je řešením minimalizační úlohy bod, kde kvadratická forma nabývá nulové hodnoty.

Analogicky budeme postupovat v případě, kdy zneutralizujeme vliv první poruchové složky signálu z . Výsledek je na obr2.3. V tomto případě nám opět vyšla nenulová hodnota kvadratické formy.

Můžeme tedy shrnout výsledek naší detekce. V čase $t = 0$ na systém působila porucha z_2 (porucha senzoru), o působení poruchy z_1 nemůžeme nic říct.



Obrázek 2.2: Minimální norma pro $Q = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$



Obrázek 2.3: Minimální norma pro $Q = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$

Kapitola 3

Rekurzivní algoritmus minimalizace

3.1 Motivace

V části 1.7 jsme popsali principiální řešení našeho problému. Na první pohled je ovšem zřejmé, že v takové formě, v jaké bylo řešení popsáno, není vhodné k praktickému použití. Je to způsobeno tím, že v každém časovém okamžiku, to znamená vždy, když naměříme nějaká data, se nám zvětšuje dimenze prostoru, na kterém hledáme řešení. V našem případě se tedy neustále zvětšuje matice omezení H . Proto bychom potřebovali, aby na konci každého kroku našeho algoritmu byla velikost příslušné matice stejná jako na začátku.

3.2 Kvazidiagonální tvar lineárních omezení

K zachování konstantní velikosti matice H po každém kroku se potřebujeme v každém kroku zbavit tolka omezení (tj. řádků matice H), kolik měříme výstupů (měřená data), a tolka neznámých (tj. sloupců matice), kolik je poruchových signálů (počet složek vektoru z). K tomu ovšem potřebujeme, aby neznámé, kterých se zbavíme v daném kroku, už v dalších omezení nevystupovaly. Tento požadavek ovšem tvar, ve kterém jsme omezení uvedli, nesplňuje. Proto bychom potřebovali, aby matice H byla tzv. kvazidiagonální matice. Nejprve si tedy definujme tento pojem.

Matice H je kvazidiagonální matice $(m, n) \times K$, jestliže její velikost je $(m \cdot K, n + (K - 1) \cdot n(z))$ a platí:

- $H(i, j) = 0$ pro $i > m, 1 \leq j \leq n(z)$
- $H(i, j) = 0$ pro $j > n, 1 \leq i \leq m$

- submatice $H(m+1 : m \cdot K, n(z)+1 : n+K \cdot n(z))$ je také kvazidiagonální matice $(m, n) \times (K - 1)$.

Pro lepší představu významu tohoto pojmu, znázorněme kvazidiagonální matici graficky

$$H = \begin{bmatrix} R & O & O & O & O & O \\ O & R & O & O & O & O \\ O & O & R & O & O & O \\ O & O & O & R & O & O \\ O & O & O & O & R & O \\ O & O & O & O & O & R \end{bmatrix} \quad (3.1)$$

V této ukázce je $K = 6$, matice R s obecnými prvky je rozměru právě (m, n) a $n(z)$ je rovno počtu takových sloupců matice R , pod kterými jsou v matici H již pouze samé nuly. Je to tedy dimenze nulové matice O . Význam takového to tvaru matice H pro náš problém je evidentní. Pro každou neznámou (odpovídající nějakému sloupci) existuje takový řádek i , že pro všechny řádky $j > i$ platí, že se v nich příslušná proměnná nevyskytuje. Otázkou tedy zůstává, zda je možné matici omezení převést do kvazidiagonálního tvaru.

Odpověď dostaneme velmi jednoduše, uvědomíme-li si, co fyzikálně znamená skutečnost, že omezení generována dynamickým systémem jsou v takovém to tvaru. Zřejmě to znamená, že vstupní signály, které na systém působily v čase t , ovlivňují výstup v čase $t, t + 1, \dots, t + \tau$, ale v čase větším než $(t + \tau)$ už nikoliv. Ale toto je přesně charakteristika dynamických systémů s konečnou dynamikou, přičemž číslo τ charakterizuje právě tuto dynamiku (jinak také řád systému). A jelikož se zabýváme pouze systémy s konečnou dynamikou, je odpověď na otázku, zda lze převést matici omezení do kvazidiagonálního tvaru, kladná. Stačí tedy vyřešit, jak takovýto tvar matice omezení získat.

Je nasnadě, že k cíli povede, pokud namísto stavového popisu systému, použijeme popis, který přímo explicitně vyjadřuje hodnoty současného výstupu jako funkci minulých hodnot vstupů a výstupu. Za popis systému tedy použijeme lineární diferenční rovnici, kterou analogicky jako u stavového popisu rozšíříme o poruchové (neměřitelné) signály z . Ukažme, jak tento popis převedeme na soustavu lineárních omezení proměnné z v kvazidiagonálním tvaru.

Lineární diferenciální rovnice pro systém s jedním vstupem a jedním výstupem rozšířená o neměřitelný vstup z vypadá takto

$$y(k+n) + a_{n-1}y(k+n-1) + \dots + a_0y(k) = \\ b_n u(k+n) + \dots + b_0 u(k) + c_n z(k+n) + \dots + c_0 z(k) \quad (3.2)$$

To lze zkráceně zapsat

$$y(k+n) + \sum_{i=0}^{n-1} a_i y(k+i) - \sum_{i=0}^n b_i u(k+i) = c_0 z(k) + \dots + c_n z(k+n)$$

Analogicky pro systém s více vstupy (včetně těch neměřitelných) a více výstupy (těch je m)

$$y_j(k+n) + \sum_{i=0}^{n-1} a_i^j y_j(k+i) - \sum_{i=0}^n b_i^j u(k+i) = \\ c_0^j z(k) + \dots + c_n^j z(k+n), \quad j = 1, \dots, m, \quad (3.3)$$

kde u a z jsou sloupcové vektory rozměru $n(u)$, resp. $n(z)$ a b^j a c^j jsou řádkové vektory stejných rozměrů (tedy $n(u)$, resp. $n(z)$).

Označme levou stranu rovnice (3.3) $r_j(k)$. Potom můžeme pro jeden časový okamžik $k = 0$ přepsat rovnici (3.3) do maticového tvaru

$$\begin{bmatrix} r_1(0) \\ r_2(0) \\ \vdots \\ r_m(0) \end{bmatrix} = \begin{bmatrix} c_0^1 & c_1^1 & \dots & c_n^1 \\ c_0^2 & c_1^2 & \dots & c_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ c_0^m & c_1^m & \dots & c_n^m \end{bmatrix} \cdot z_n^0, \quad (3.4)$$

zkráceně $r(0) = R \cdot z_n^0$,

přičemž matice R má rozložení $(m, (n+1) \cdot n(z))$.

Potom pro $k = 0, 1, \dots, T$ dostaneme

$$h = \begin{bmatrix} r(0) \\ r(1) \\ \vdots \\ r(T) \end{bmatrix} = H \cdot z_{T+n}^0$$

a matice H má strukturu jako v 3.1, je to tedy kvazidiagonální matice rozložení $(m, (n+1) \cdot n(z)) \times (T+1)$.

3.3 Rekurzivní algoritmus

Nyní můžeme přistoupit k popsání rekurzivního algoritmu, který řeší náš problém. Zopakujme tedy, že naším úkolem je minimalizovat kvadratickou formu $z^T Q z$, za lineárních omezení $H z = h$, přičemž matice H je v kvazidiagonálním tvaru. Celý algoritmus budeme průběžně demonstrovat na našem příkladu. Nejprve uvedeme definici dvou pojmu, které budeme v následující části potřebovat.

3.3.1 Pomocné pojmy

QR rozklad matice X je rozklad, který generuje horní trojúhelníkovou matici R , stejně velikosti jako matice X , a ortonormální matici Q takovou, že platí $X = QR$.

Choleskyho faktorizace je rozklad libovolné pozitivně definitní matice X do tvaru $X = X_c^T X_c$, kde X_c je reálná horní trojúhelníková matice. Takto definovaná Choleskyho faktorizace se označuje jako standardní. Její vlastností je, že až na znaménko každého řádku (proto se uvažují všechny diagonální prvky kladné) je jednoznačná. Je-li matice X pozitivně semidefinitní, pak Choleskyho faktorizace není jednoznačná. Na diagonále matice X_c se vyskytnou nulové prvky. Na těchto řádcích, kde diagonální prvek je nulový, nejsou ostatní prvky definovány. Proto je můžeme zvolit. Zvolíme-li je jako nulové, budeme pak takovou faktorizaci označovat jako zobecněnou Choleskyho faktorizaci.

3.3.2 Úprava minimalizované kvadratické formy

Nyní převedeme naši kvadratickou formu do tvaru, který je vhodný pro numerickou optimalizaci

$$z^T Q z = z^T Q_c^T Q_c z = (Q_c z)^T Q_c z = \|Q_c z\|^2 = \|F x\|^2,$$

kde $F = \begin{bmatrix} Q_c & 0 \\ 0 & \dots & 0 \end{bmatrix}$ a $x = \begin{pmatrix} z \\ 1 \end{pmatrix}$. Počítání s Choleskyho faktorem

matice Q má tu výhodu, že ani díky zaokrouhlovacím chybám neztrácí matice Q svojí pozitivní (semi)definitnost, neboť součin $X^T X$ libovolné matice X je vždy pozitivně (semi)definitní. Proto se také Choleskyho faktor někdy nazývá maticová odmocnina. Vektor z jsme rozšířili o absolutní člen z toho důvodu, že omezení mají tvar nehomogenních lineárních rovnic.

Označme q čtvercovou matici rozměru $n(z)$, která je Choleskyho faktorem matice jádra kvadratické formy. Typicky bude tato matici diagonální, jak bylo diskutováno v předchozí části. Na začátku celého algoritmu zvolíme čtvercovou matici Q_c o velikosti $(n+1) \cdot n(z)$, kde n je řád systému a $n(z)$ je počet složek signálu z (počet poruchových signálů).

$$Q_c = \begin{bmatrix} q & O & \cdots & O \\ O & q & \ddots & \vdots \\ \vdots & \ddots & \ddots & O \\ O & \cdots & 0 & q \end{bmatrix}$$

V našem příkladu by to byla matici 6×6 . Z této matice vytvoříme matici F , tak že k matici Q_c přidáme jeden nulový řádek a jeden nulový sloupec. Velikost takto získané matice definuje dimenzi prostoru, v kterém budeme řešit náš problém.

3.3.3 Minimální hodnota kvadratické formy

Jelikož nás bude v každém kroku algoritmu zajímat, jaká je minimální hodnota kvadratické formy definované maticí F , ukažme si jak lze nejjednodušeji tuto hodnotu získat. Provedeme proto následující úpravu

$$\begin{aligned} \|F \begin{bmatrix} z \\ 1 \end{bmatrix}\|^2 &= \left\| \begin{bmatrix} F_z & F_{z,1} \\ O & f_1 \end{bmatrix} \begin{bmatrix} z \\ 1 \end{bmatrix} \right\|^2 = \\ &= \left\| \begin{pmatrix} F_z z + F_{z,1} \\ f_1 \end{pmatrix} \right\|^2 = \|F_z z + F_{z,1}\|^2 + f_1^2 \end{aligned}$$

Potom pro z , minimalizující tento výraz, dostaneme

$$F_z z + F_{z,1} = 0 \tag{3.5}$$

Předpokládejme, že matice F je horní trojúhelníková, přičemž pokud má na i -té pozici na diagonále nulu, pak je celý i -tý řádek nulový. Jinými slovy, matice F je zobecněným Choleskyho faktorem matice M ($M = F^T F$). Jak tento předpoklad zajistíme, popíšeme později. Potom rovnici (3.5) můžeme splnit. Kdybychom chtěli výpočet optimálního z provést, počítali bychom jednotlivé prvky vektoru z odspodu. Přičemž, byl-li by i -tý řádek matice F_z nulový, zvolili bychom i -tou složku vektoru z libovolně. Minimální hodnota kvadratické formy tedy je

$$\left\| F \begin{bmatrix} z \\ 1 \end{bmatrix} \right\|^2 = f_1^2,$$

tedy druhá mocnina posledního prvku na diagonále matice F . Na začátku je tedy hodnota kvadratické normy nulová (matice F je v diagonálním tvaru a na posledním místě je nula), což je logické, neboť nemáme žádná omezení a každá pozitivně (semi)definitní kvadratická forma má v minimu hodnotu nula.

3.3.4 Zahrnutí omezení do matice F

Doposud popsaná část algoritmu byla pouze jakási inicializace. Nyní přistupme k vlastní rekurzivní části. Rekurzivní část našeho algoritmu začíná získáním naměřených dat (výstupů). Těch je, jak již jsme dříve označili, právě m . Nyní jde tedy o to, jak tato omezení zahrnout do jádra naší kvadratické formy, tedy do matice F .

Provedeme to tak, že m proměnných vyjádříme jako kombinaci těch zbývajících. Předpokládáme, že $m < n(z)$, neboli že počet omezení (měřené výstupy) je menší než počet složek vektoru z (počet poruchových signálů). Je patrné, že tento předpoklad není nikterak omezující, neboť kdyby nebyl splněn, tak by to znamenalo, že soustava lineárních omezení je buď řešitelná jednoznačně, nebo nemá řešení vůbec. Ani v jednom z těchto případů by pak nemělo smysl provádět nějakou minimalizaci, neboť množina, na které bychom tuto minimalizaci prováděli, by byla buď jednoprvková, nebo prázdná. Poznamenejme ještě, že v praxi je tato nerovnost splněna s velkou rezervou, neboť počet veličin, které můžeme měřit, je často velmi omezen.

Dále budeme předpokládat, že řádky matice R jsou lineárně nezávislé, tedy že hodnost matice R je m . Tento předpoklad lze považovat za oprávněný, uvědomíme-li si, že jednotlivé řádky matice R reprezentují jednotlivá měření, tedy jednotlivé výstupy naší soustavy. Lineární závislost řádků matice R by pak znamenala, že existuje nějaký výstup, který lze vyjádřit jako lineární kombinaci ostatních výstupů. Je zřejmé, že takový výstup je nadbytečný, a proto ho nemá smysl měřit.

Máme tedy m lineárně nezávislých omezení, které chceme zahrnout do kvadratické formy definované maticí F . Jak jsme uvedli výše, uděláme to tak, že m proměnných vyjádříme jako lineární kombinaci zbývajících proměnných. Problém je, kterých m proměnných zvolit. Nejjednodušší by bylo vybrat m složek vektoru $z(t)$, kde t je dle značení v rovnici (3.4) $i - 1$, kde i je číslo právě prováděné iterace algoritmu. Jinak řečeno, vybrali bychom m složek nejstaršího vektoru z , který se vyskytuje v minimalizované kvadratické formě. Tento výběr by měl tu výhodu, že bychom vyjádřili proměnné, které by se již nemohly vyskytovat v budoucích omezení. Abychom ale tyto proměnné mohli vyjádřit, musela by matice tvořená sloupcem matice R příslušející těmto proměnným být invertovatelná. To ale obecně splněno být nemusí, lépe ře-

čeno typicky ani nebývá. Proto musíme vybrat takové proměnné, které toto splňují. To provedeme tak, že postupně budeme vybírat proměnné, a to takto: pro j -tou vybranou proměnnou platí, že je různá od proměnné $j - 1, j - 2, \dots, 1$ a příslušný sloupec matice R má na j -té pozici nenulový prvek. Tyto dvě vlastnosti nám určují množinu přípustných j -tých proměnných, přičemž vybereme tu, jejíž sloupec má nejmenší index (je v matici R nejvíce vlevo).

Máme-li nyní m proměnných, které můžeme vyjádřit pomocí těch zbývajících, přesuneme sloupce matice R odpovídající těmto proměnným na začátek této matice. To můžeme provést vynásobením matice R zprava maticí T_s . Přičemž matice T_s je ortogonální matice, jejíž sloupce jsou tvořeny nulovými vektory s jedním jednotkovým prvkem. Potom, chceme-li přesunout j -tý sloupec matice R na i -tou pozici, bude mít i -tý sloupec matice T_s jednotkový prvek na j -té pozici.

Naznačme tedy tuto transformaci souřadnic (tedy vyjádření m proměnných jako lineární kombinaci těch zbývajících). Nejprve formálně upravíme soustavu omezení pomocí rozšíření vektoru z_{t+n}^t o absolutní člen.

$$r(t) = Rz_{t+n}^t \longrightarrow \overline{R}(t)\overline{z}_{t+n}^t = 0,$$

$$\overline{R}(t) = \begin{bmatrix} R & -r(t) \end{bmatrix}, \overline{z}_{t+n}^t = \begin{bmatrix} z_{t+n}^t \\ 1 \end{bmatrix}$$

Nyní přesuneme vybrané sloupce vynásobením maticí T_s a nově vzniklou matici rozdělíme na dvě submatice

$$\begin{aligned} \overline{R}_S &= \overline{R} \cdot T_S \\ \overline{R}_S &= \begin{bmatrix} \overline{R}_{S1} & \overline{R}_{S2} \end{bmatrix}, \overline{R}_{S1} - \text{matica } (m, m) \end{aligned}$$

Jednoduchými úpravami postupně dostaneme

$$\begin{aligned} \overline{R}\overline{z}_{t+n}^t &= 0 \\ \overline{R}T_S T_S^T \overline{z}_{t+n}^t &= 0, \text{ neboť } T_S T_S^T = I \\ T_S^T \overline{z}_{t+n}^t &= \begin{bmatrix} -\text{inv}(\overline{R}_{S1}) \cdot \overline{R}_{S2} \\ I \end{bmatrix} \cdot \overline{z}_{t-+n}^{t-} \end{aligned} \tag{3.6}$$

$$\begin{aligned} \overline{z}_{t+n}^t &= T_S \cdot \begin{bmatrix} -\text{inv}(\overline{R}_{S1}) \cdot \overline{R}_{S2} \\ I \end{bmatrix} \cdot \overline{z}_{t-+n}^{t-} \\ \overline{z}_{t+n}^t &= A(t) \overline{z}_{t-+n}^{t-} \end{aligned} \tag{3.7}$$

Vektor \bar{z}_{t-+n}^{t-} značí vektor \bar{z}_{t+n}^t bez m vybraných proměnných. V tomto tvaru (3.7) již můžeme omezení jednoduše zahrnout do jádra kvadratické formy, což ukážeme v další kapitole. Tím dojde k tomu, že kvadratická forma bude závislá pouze na proměnných obsažených v \bar{z}_{t-+n}^{t-} . Tam ale nemusí být některé proměnné, které se budou vyskytovat v dalších omezeních. Proto musíme do matice omezení v dalším kroku rovněž zahrnout omezení vyjádřená v tomto kroku. Za tímto účelem provedeme následující úpravy.

$$\begin{aligned} A(t) &= \begin{bmatrix} A_1 & a_2 \\ a_3^T & 1 \end{bmatrix} \\ \bar{z}_{t+n+1}^t &= \begin{bmatrix} A_1 & O & a_2 \\ O & I & o \\ a_3^T & o^T & 1 \end{bmatrix} \begin{bmatrix} \bar{z}_{t-+n}^{t-} \\ \bar{z}(t+n+1) \end{bmatrix} = A'(t) \bar{z}_{t-+n+1}^{t-} \\ \bar{z}_{t+n+1}^{t+1} &= A'_R(t) \bar{z}_{t-+n+1}^{t-} = A_R(t) \bar{z}_{t-+n+1}^{t-+1} \end{aligned}$$

Přičemž matice A'_R vznikne z matice A' jednoduše pouhým odebráním prvních $n(z)$ řádků. Matice A_R pak vznikne z matice A'_R odebráním tolika prvních sloupců, kolik je prvků ve vektoru $z(t^-)$. To můžeme udělat proto, protože tyto sloupce jsou nulové. Nulovost těchto sloupců je důsledek skutečnosti, že jsme v (3.6) vyjadřovali proměnné jako funkce minulých, popř. současných proměnných. To vyplývá z toho, že jsme při výběru proměnných, které budeme eliminovat, brali vždy ty, jejichž sloupce byly v matici R nejvíce vlevo (samozřejmě z těch sloupců, které měly na příslušném řádku nenulový prvek, jak jsme podrobněji popsali výše). A pro omezení v čase $t+1$ konečně platí

$$\begin{aligned} \bar{R} \bar{z}_{t+n+1}^{t+1} &= 0 \\ \bar{R} A_R(t) \bar{z}_{t-+n+1}^{t-+1} &= 0. \end{aligned}$$

A nyní se celý postup opakuje, takže hledáme vhodné proměnné pro eliminaci s tím, že místo matice \bar{R} máme matici $\bar{R} \cdot A_R$. Analogicky tedy dostaneme matici $A(t+1)$ a platí

$$\bar{z}_{t-+n+1}^{t-+1} = A(t+1) \bar{z}_{t--+n+1}^{t--+1}.$$

Vektor $\bar{z}_{t--+n+1}^{t--+1}$ opět značí vektor \bar{z}_{t-+n+1}^{t-+1} bez m vybraných proměnných. Jelikož ale potřebujeme vztah mezi \bar{z}_{t+n+1}^{t+1} a $\bar{z}_{t--+n+1}^{t--+1}$, musíme vynásobit matici $A(t+1)$ maticí $A_R(t)$, neboť platí

$$\bar{z}_{t+n+1}^{t+1} = A_R(t) \bar{z}_{t^-+n+1}^{t^-+1} = A_R(t) A(t+1) \bar{z}_{t--+n+1}^{t--+1}.$$

Dosaďme tedy transformaci proměnných do kvadratické normy.

$$\begin{aligned} \min_{z, Rz=0} \|F\bar{z}_{t+n}^t\|^2 &= \min_z \left\| F \cdot A(t) \cdot \bar{z}_{t^-+n}^{t^-} \right\|^2 \\ &= \min_z \left\| F_x \bar{z}_{t^-+n}^{t^-} \right\|^2. \end{aligned} \quad (3.8)$$

Nyní provedeme QR rozklad matice F_x

$$\begin{aligned} F_x &= Q_f R_f, \quad Q_f^T Q_f = I \\ \left\| F_x \bar{z}_{t^-+n}^{t^-} \right\|^2 &= \left\| Q_f R_f \bar{z}_{t^-+n}^{t^-} \right\|^2 = \left(\bar{z}_{t^-+n}^{t^-} \right)^T R_f^T Q_f^T Q_f R_f \left(\bar{z}_{t^-+n}^{t^-} \right) = \\ &= \left(\bar{z}_{t^-+n}^{t^-} \right)^T R_f^T R_f \left(\bar{z}_{t^-+n}^{t^-} \right) = \left\| R_f \bar{z}_{t^-+n}^{t^-} \right\|^2 = \left\| R_x \bar{z}_{t^-+n}^{t^-} \right\|^2, \end{aligned} \quad (3.9)$$

kde matice R_x je matice R_f bez m posledních nulových řádků, neboli je to její největší čtvercová submatice obsahující prvek $(1, 1)$.

Výše popsaný způsob zahrnutí m lineárně nezávislých omezení do jádra kvadratické formy je teoreticky zcela korektní a univerzálně použitelný, ale může někdy narazit na numerické problémy. Proto může být pro praktickou realizaci vhodnější formálně přidat k stávajícím proměnným m nových proměnných. Ty začleníme do omezení tak, abychom právě tyto proměnné mohli jednoduše vyjádřit. To znamená, že matici R rozšíříme zleva o jednotkovou matici. Abychom ale neporušili hodnoty původních rovnic musíme zajistit, aby hodnoty nově přidaných proměnných byly nulové. To zajistíme tak, že váhy těchto proměnných v jádru kvadratické formy zvolíme o několik řádů vyšší než jsou váhy „reálných“ proměnných z . To způsobí, že při minimalizaci takovéto kvadratické formy dojde prakticky k vynulování uměle přidaných proměnných. Označíme-li nově přidané proměnné např. e , můžeme celý postup zkráceně zapsat

$$\begin{aligned}
\bar{R}(t)\bar{z}_{t+n}^t = 0 \rightarrow \begin{bmatrix} E & \bar{R} \end{bmatrix} \begin{bmatrix} e \\ \bar{z}_{t+n}^t \end{bmatrix} = 0 \\
\begin{bmatrix} e \\ \bar{z}_{t+n}^t \end{bmatrix} = \begin{bmatrix} -\bar{R} \\ E \end{bmatrix} \bar{z}_{t+n}^t = A\bar{z}_{t+n}^t \\
BB = \begin{bmatrix} bb & & \\ & \ddots & \\ & & bb \end{bmatrix}, bb = 10000 \\
\min_z \|F\bar{z}_{t+n}^t\|^2 \rightarrow \min_{e,z} \left\| \begin{bmatrix} BB & O \\ O & F \end{bmatrix} \begin{bmatrix} e \\ \bar{z}_{t+n}^t \end{bmatrix} \right\|^2 = \\
= \min_z \left\| \begin{bmatrix} BB & \\ & F \end{bmatrix} A\bar{z}_{t+n}^t \right\|^2 = \min_z \|R_x\bar{z}_{t+n}^t\|^2.
\end{aligned}$$

V kapitole 4.5.3 budeme potřebovat zahrnout omezení do kvadratické formy, která bude v argumentu hustoty pravděpodobnosti. Použijeme stejný „trik“ s přidáním proměnných, a to přesto, že zde nebudeme fakticky provádět žádnou minimalizaci. Budeme integrovat hustotu pravděpodobnosti přes lineární podprostor a přidání nových proměnných hodnotu tohoto integrálu nezmění, neboť velké koeficienty v jádru kvadratické formy budou představovat zanedbatelně malý rozptyl nově přidaných proměnných, lépe řečeno náhodných proměnných. To znamená, že nová vícerozměrná hustota pravděpodobnosti náhodného vektoru $\begin{pmatrix} e \\ z \end{pmatrix}$ bude nenulová prakticky jen pro $e = 0$, což je přesně to, co potřebujeme.

3.3.5 Úprava na zobecněný Choleskyho faktor

Pokud bude mít matice R_x někde na diagonále nulový prvek, musíme ještě tuto matici upravit do tvaru, který odpovídá zobecněnému Choleskyho faktoru, jak jsme dříve definovali. To můžeme provést tak, že prvky na příslušném řádku, kde je na diagonále nulový prvek, vynulujeme. Ostatní prvky ležící pod daným řádkem přepočítáme podle algoritmu výpočtu prvků Choleskyho rozkladu (např. v [5]). Jednotlivé prvky počítáme po řádcích odshora dolů. Nejprve spočítáme prvek na diagonále

$$R_x(i,i) = \sqrt{M(i,i) - \sum_{k=1}^{i-1} R_x^2(k,i)},$$

kde $M = R_x^T R_x$ a potom prvky vpravo od diagonály

$$R_x(i, j) = \frac{1}{R_x(i, i)} \left[M(i, j) - \sum_{k=1}^{i-1} R_x(k, i) R_x(k, j) \right], j > i$$

3.3.6 Vlastní minimalizace

Vzhledem k tomu, že jsme již zahrnuli všechna omezení do kvadratické formy a matice této formy je horní trojúhelníková, platí, že minimální hodnota této formy je rovna druhé mocnině posledního prvku na diagonále. Proto bychom v tuto chvíli mohli uzavřít celý jeden krok iterace algoritmu a po nezbytné úpravě matice R_x na matici F (rozšíření matice o $n(z)$ řádků a sloupců) začít další krok (nebo-li zpracovat data z dalšího časového okamžiku). Ovšem tím by nám v každém kroku algoritmu vzrostla dimenze matice F o $(n(z) - m)$. Neboť v každém kroku nám přibude $n(z)$ proměnných (jeden vektor z) a díky omezením odstraníme m proměnných, jak jsme popsali v předchozí části.

Ale přitom hodnota těch proměnných, které odpovídají zbylým složkám nejstaršího vektoru z , které nebyly odstraněny při dosazení omezení do kvadratické normy, je již v daném okamžiku určena, neboť tyto proměnné se nemůžou vyskytovat v dalších omezeních. Nabývají totiž takových hodnot, aby minimalizovali danou kvadratickou normu. Pro hodnotu kvadratické normy daného vektoru x platí

$$\|x\|_2^2 = \langle x, x \rangle = x^T x = x_1^2 + x_2^2 + \dots + x_n^2$$

Z toho vyplývá, že minimum kvadratické normy dosáhneme, jestliže vynu-
lujeme jednotlivé složky vektoru x , v našem případě vektoru $R_x \bar{z}$. Rozložíme-li matici R_x na jednotlivé submatice, můžeme tento vektor vyjádřit takto

$$R_x \bar{z}_{t^-+n} = \begin{bmatrix} R_{x1} z(t^-) + R_{x2} \bar{z}_{t^-+n}^{t^-+1} \\ R_{x3} \bar{z}_{t^-+n}^{t^-+1} \end{bmatrix},$$

přičemž $R_x = \begin{bmatrix} R_{x1} & R_{x2} \\ O & R_{x3} \end{bmatrix}$ a R_{x1} je rozměru $(n(z) - pe, n(z) - pe)$,

kde pe je počet složek vektoru $z(t)$, které jsme eliminovali.

Je vidět, že proměnné, které se již nemohou vyskytovat v budoucích omezeních, se vyskytují pouze v prvních $n(z) - pe$ složkách vektoru $R_x \bar{z}$. Jejich hodnoty jsou tedy určeny rovnicí

$$R_{x1} z(t^-) + R_{x2} \bar{z}_{t^-+n}^{t^-+1} = 0 \quad (3.10)$$

Tato rovnice je vlastně formálně shodná s omezeními, které dostáváme měřením výstupů systému. Proto také princip, který použijeme pro dosazení těchto omezení do kvadratické normy, bude formálně shodný. Matice R_{x1} odpovídá matici \bar{R}_{S1} a matice R_{x2} odpovídá matici \bar{R}_{S2} . Jediný rozdíl je v tom, že již nemůžeme vyloučit singulárnost matice \bar{R}_{x1} , to jest matice, kterou bychom potřebovali invertovat. Ale jelikož jsme si matici R_x upravili do tvaru zobecněného Choleskyho faktoru, víme, že matice R_{x1} má na diagonále buď nenulové prvky, anebo je příslušný řádek celý nulový (rovněž příslušný řádek matice R_{x2} je nulový). Proto víme, že řešení rovnice (3.10) existuje, přičemž složky vektoru $z(t^-)$, které odpovídají nulovým řádkům, můžeme volit libovolně.

Tuto volbu můžeme formálně provést tak, že matici R_{x1} v rovnici (3.10) nahradíme maticí R'_{x1}

$$R'_{x1}z(t^-) + R_{x2}\bar{z}_{t^-+n}^{t^-} = 0,$$

přičemž matice R'_{x1} je shodná s maticí R_{x1} až na nulové prvky na diagonále, které jsou nahrazeny libovolným nenulovým číslem (uvažujme například číslem 1). Potom je matice R'_{x1} invertovatelná a analogicky s rovnicí (3.7) můžeme psát

$$\bar{z}_{t^-+n}^{t^-} = \begin{bmatrix} -\text{inv}(R'_{x1}) \cdot R_{x2} \\ I \end{bmatrix} \cdot \bar{z}_{t^-+n}^{t^-+1} \quad (3.11)$$

Tento výsledek dosadíme do vztahu pro kvadratickou formu (3.9) opět analogicky s rovnicí (3.8)

$$\begin{aligned} \min_z \|R_x \bar{z}_{t^-+n}^{t^-}\|^2 &= \min_z \left\| R_x \cdot \begin{bmatrix} -\text{inv}(R'_{x1}) \cdot R_{x2} \\ I \end{bmatrix} \cdot \bar{z}_{t^-+n}^{t^-+1} \right\|^2 \\ &= \min_z \|F_m \bar{z}_{t^-+n}^{t^-+1}\|^2 \end{aligned} \quad (3.12)$$

Čistě mechanicky bychom mohli opět provést QR rozklad matice F_m , a poté odstranit posledních $(n(z) - pe)$ nulových řádků. Tím bychom dostali čtvercovou horní trojúhelníkovou matici, označíme ji R_m . Ukažme ale, že tato matice je totožná se submaticí R_{x3} matice R_x . Upravíme proto součin matic v jádru kvadratické normy v rovnici (3.12)

$$\begin{aligned} \begin{bmatrix} R_{x1} & R_{x2} \\ O & R_{x3} \end{bmatrix} \cdot \begin{bmatrix} -\text{inv}(R'_{x1}) \cdot R_{x2} \\ I \end{bmatrix} &= \\ &= \begin{bmatrix} -R_{x1} \cdot \text{inv}(R'_{x1}) \cdot R_{x2} + R_{x2} \\ 0 + R_{x3} \end{bmatrix} = \begin{bmatrix} O \\ R_{x3} \end{bmatrix} \end{aligned} \quad (3.13)$$

$$\begin{bmatrix} O \\ R_{x3} \end{bmatrix} \cdot \bar{z}_{t^-+n}^{t^-+1} = R_{x3} \bar{z}_{t^-+n}^{t^-+1}$$

A jelikož matice R_{x3} je ve tvaru zobecněného Choleskyho faktoru matice $R_{x3}^T R_{x3}$ (neboť matice R_x je zobecněným Choleskyho faktorem matice $R_x^T R_x$) nemusíme již provádět žádné další maticové úpravy. Je tedy vidět, že při praktickém výpočtu stačí výpočty popsané rovnicemi (3.11),(3.12),(3.13) nahradit přiřazením $R_m = R_{x3}$. Rovnice (3.11),(3.12),(3.13) provádějí pouze odvození tohoto vztahu.

3.3.7 Závěrečná úprava

Tím jsme se dostali na konec jedné iterace našeho algoritmu. Před zahájením další iterace musíme ještě rozšířit matici R_m na původní velikost matice F , abychom mohli zpracovat data z dalšího časového okamžiku. Tedy

$$\begin{aligned} \min_{z_{t+n}^{t+1}} \|R_m \bar{z}_{t+n}^{t+1}\|^2 &= \min_{z_{t+n+1}^{t+1}} \left\| \begin{pmatrix} R_m \bar{z}_{t+n}^{t+1} \\ q z(t+n+1) \end{pmatrix} \right\|^2 = \\ &= \min_{z_{t+n+1}^{t+1}} \|F \bar{z}_{t+n+1}^{t+1}\|^2 \\ R_m &= \left[\begin{array}{c|c} R_{m_1} & R_{m_2} \\ \hline R_{m_3} & r \end{array} \right], \quad F = \left[\begin{array}{ccc|cc} & & & O & \\ R_{m_1} & & & \vdots & R_{m_2} \\ & & O & & \\ \hline O & \cdots & O & q & O \\ \hline & & & 0 & r \end{array} \right]. \end{aligned}$$

Poznamenejme ještě, že přestože matice F nemusí být horní trojúhelníková (ale ve většině případů se dá předpokládat, že je, neboť q bývá diagonální matice), je minimální hodnota této kvadratické normy stále dána druhou mocninou posledního prvku (v poslední rovnici označen jako r), neboť jsme původní normu rozšířili o $z(t+n+1)^T q^T q z(t+n+1)$. Tento přírůstek lze volbou $z(t+n+1) = 0$ minimalizovat na nulovou hodnotu.

3.3.8 Shrnutí

Na závěr shrneme celý průběh jedné iterace našeho algoritmu tím, že ukážeme, jak se mění velikost (rozměr) matice v jádru kvadratické normy.

1. $F(t) - d(t) \times d(t)$, $d(0) = (n+1) \cdot n(z) + 1$
2. $F_x(t) - d(t) \times (d(t) - m)$ -zahrnutí omezení

3. $R_x(t) - (d(t) - m) \times (d(t) - m)$ -trojúhelníkování a odstranění m nulových řádků
4. $F_m(t) - (d(t) - m) \times (d(t) - m - n(z) + pe(t))$ -minimalizace normy přes proměnné, které již nemohou vystupovat v budoucích omezeních
5. $R_m(t) - (d(t) - m - n(z) + pe(t)) \times (d(t) - m - n(z) + pe(t))$ -trojúhelníkování a odstranění $n(z) - pe(t)$ nulových řádků
6. $F(t+1) - (d(t) - m + pe(t)) \times (d(t) - m + pe(t)) \equiv d(t+1) \times d(t+1)$ -rozšíření o $n(z)$ nulových řádků a sloupců a o matici q na diagonále

3.4 Příklad

Pro lepší názornost ukážeme průběh celého algoritmu na našem příkladu z kapitoly 2.1. Nejprve převedeme stavový popis systému na popis diferenční rovnicí, abychom dostali omezení do kvazidiagonálního tvaru.

Nejprve vyjádříme přenosy od vstupních (měřitelných i neměřitelných) veličin na výstupní.

$$\begin{aligned} y(z) &= \left(C(zI - A)^{-1}B + D \right) u(z) + \left(C(zI - A)^{-1}F + G \right) z(z) \\ y(z) &= \frac{z-1}{z(z-0.5)} u(z) + \begin{bmatrix} \frac{z-1}{z(z-0.5)} & 1 \end{bmatrix} z(z) \end{aligned}$$

pozn.: nutno rozlišovat mezi vektorem poruch z a operátorem Z -transformací z .

Nyní již snadno zapíšeme náš systém diferenční rovnicí

$$\begin{aligned} y(t+2) - 0.5y(t+1) - u(t+1) + u(t) &= r(t) = \\ &= [-1 \ 0] z(t) + [1 \ 0.5] z(t+1) + [0 \ -1] z(t+2) \end{aligned}$$

Dle značení zavedeného v této kapitole popíšeme tedy náš systém takto

- $z(t) = \begin{pmatrix} z_1(t) \\ z_2(t) \end{pmatrix} \Rightarrow n(z) = 2$
- $R = [-1 \ 0 \ 1 \ 0.5 \ 0 \ -1] \Rightarrow m = 1, n = 2$

V čase od $t = 0$ do $t = 2$ jsme změřili výstup a vstup systému a spočítali hodnotu rezidua

$$r(0) = y(2) - 0.5y(1) - u(1) + u(0) = 2.$$

Matici q uvažujme $q = [0.9 \ 0.2]$. Potom matice v jádru normy bude nabývat v jednotlivých krocích jedné iterace algoritmu následujících hodnot

$$1. F(0) = \begin{bmatrix} 0.9 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.9 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$2. (a) \overline{R}(0) = [R \ -2], \overline{R}_1(0) = -1,$$

$$\overline{R}_2(0) = [0 \ 1 \ 0.5 \ 0 \ -1 \ -2]$$

$$(b) -\text{inv}(\overline{R}_1(0)) \cdot \overline{R}_2(0) = [0 \ 1 \ 0.5 \ 0 \ -1 \ -2]$$

$$(c) F_x(0) = F(0) \cdot \begin{bmatrix} 0 & 1 & 0.5 & 0 & -1 & -2 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} 0 & 0.9 & 0.45 & 0 & -0.9 & -1.8 \\ 0.2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.9 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$3. R_x(0) = \begin{bmatrix} -0.2000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.2728 & 0.3182 & 0 & -0.6364 & -1.2728 \\ 0 & 0 & 0.3758 & 0 & -0.5388 & -1.0776 \\ 0 & 0 & 0 & -0.9000 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.3933 & 0.5832 \\ 0 & 0 & 0 & 0 & 0 & 1.0577 \end{bmatrix}$$

$$4. F_m(0) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1.2728 & 0.3182 & 0 & -0.6364 & -1.2728 \\ 0 & 0.3758 & 0 & -0.5388 & -1.0776 \\ 0 & 0 & -0.9000 & 0 & 0 \\ 0 & 0 & 0 & 0.3933 & 0.5832 \\ 0 & 0 & 0 & 0 & 1.0577 \end{bmatrix}$$

$$5. R_m(0) = \begin{bmatrix} 1.2728 & 0.3182 & 0 & -0.6364 & -1.2728 \\ 0 & 0.3758 & 0 & -0.5388 & -1.0776 \\ 0 & 0 & -0.9000 & 0 & 0 \\ 0 & 0 & 0 & 0.3933 & 0.5832 \\ 0 & 0 & 0 & 0 & 1.0577 \end{bmatrix}$$

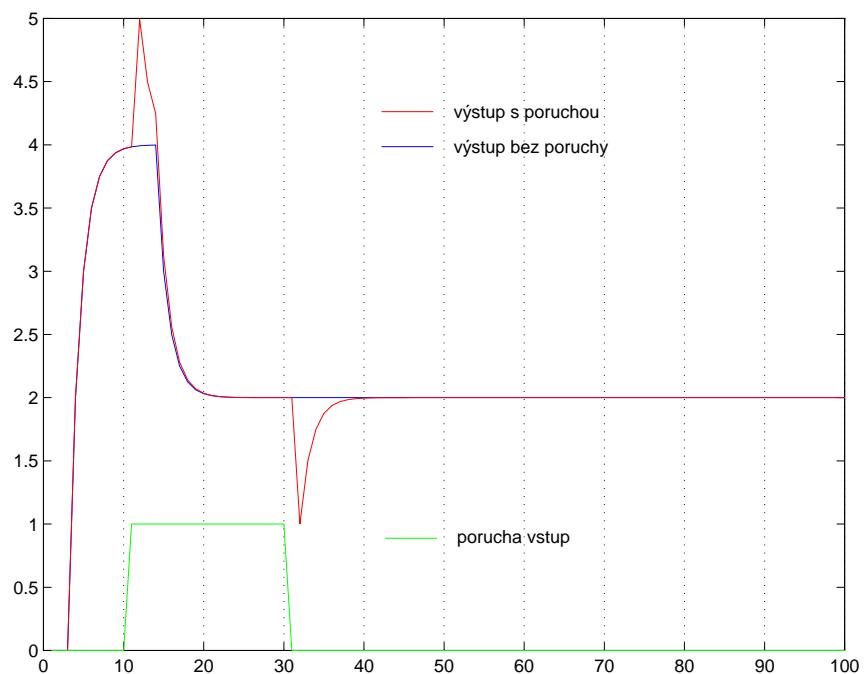
$$6. F(1) = \begin{bmatrix} 1.2728 & 0.3182 & 0 & -0.6364 & 0 & 0 & -1.2728 \\ 0 & 0.3758 & 0 & -0.5388 & 0 & 0 & -1.0776 \\ 0 & 0 & -0.9000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.3933 & 0 & 0 & 0.5832 \\ 0 & 0 & 0 & 0 & 0.9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1.0577 \end{bmatrix}$$

A minimální hodnota normy je

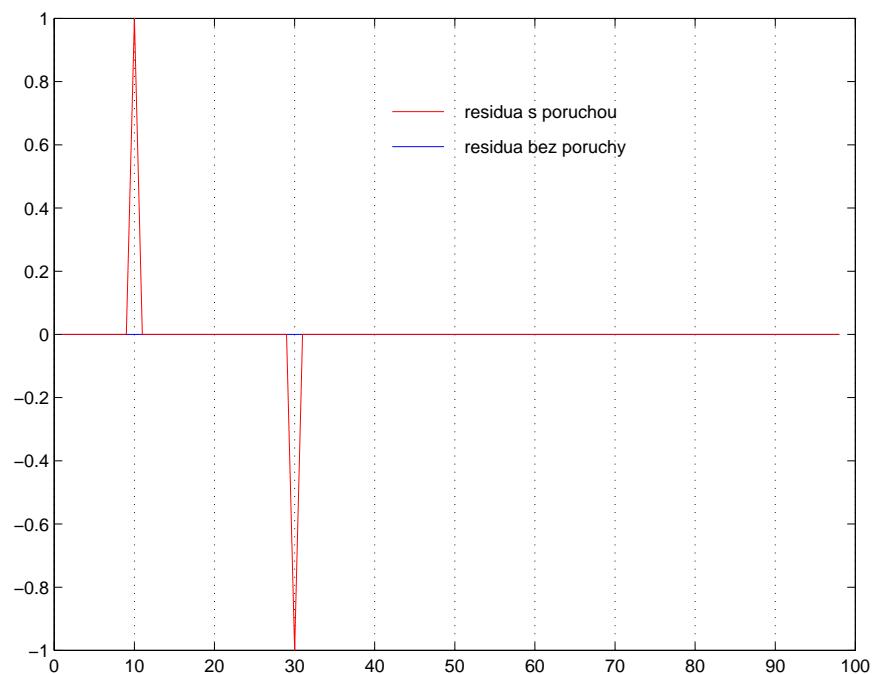
$$\min_{z\left(\begin{smallmatrix} 0 \\ 2 \end{smallmatrix}\right), R \cdot z\left(\begin{smallmatrix} 0 \\ 2 \end{smallmatrix}\right) = r(0)} \left\| F(0) \cdot z \left(\begin{matrix} 0 \\ 2 \end{matrix} \right) \right\| = R_m^2(0)(5, 5) \doteq 1.12$$

Na obr.3.1 až 3.6 jsou výsledky simulací, které jsme provedli na našem systému, přičemž jsme uvažovali poruchu působící na vstup a na výstup. Na obr.3.1 je zobrazen výstup soustavy bez působení poruchy a při působení naznačené aditivní konstantní poruchy na vstup. Porucha tedy působila od času $t = 10$ do času $t = 30$ a měla velikost 1. Na obr.3.2 je zobrazen průběh velikosti residua soustavy, opět bez poruchy (nulová hodnota) a s poruchou. A konečně na obr.3.3 je zobrazen průběh minimální normy poruchového signálu. Z tohoto průběhu je patrné, že k nárustu minimální normy, a tedy k detekci poruchy, dochází pouze v okamžiku, kdy skutečná porucha začala, resp. přestala působit. Proč tomu tak je, je zřejmé již z obr.3.1 a 3.2, kde je vidět, že přenos ze vstupu na výstup naší soustavy má derivační (diferenční) charakter, a tudíž konstantní porucha na vstupu se na výstupu neprojeví, a tudíž nemůže být ani na základě měření výstupu detekována.

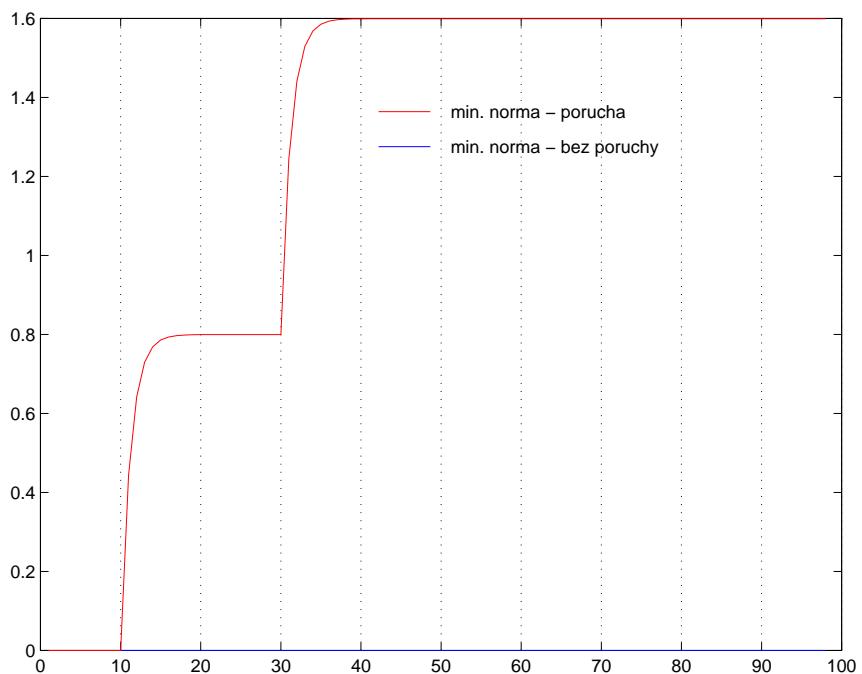
Stejné průběhy jsou zobrazeny na obr.3.4 až 3.6, ale pro poruchu působící na výstup soustavy. Zde je již porucha detekována po celou dobu jejího působení.



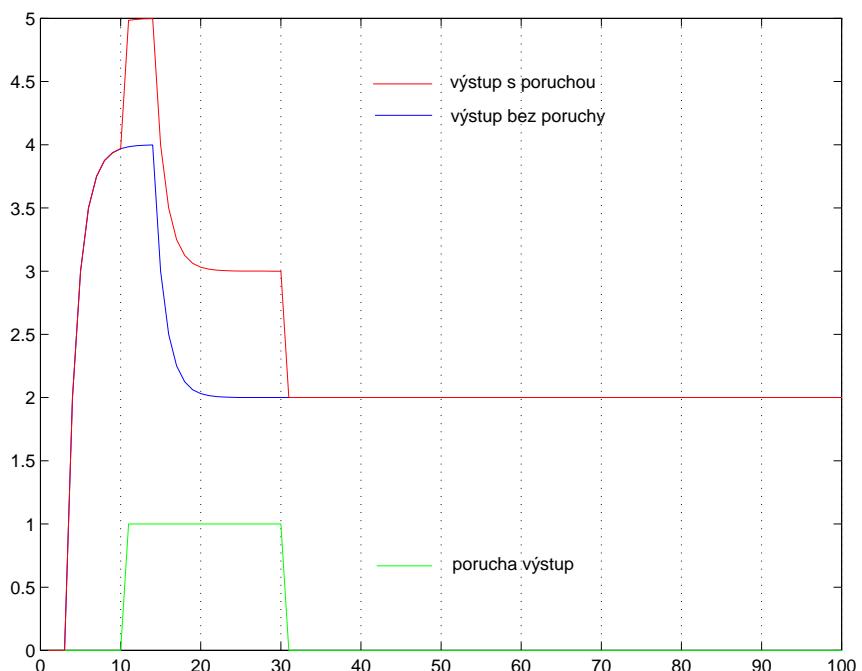
Obrázek 3.1: Výstup soustavy při poruše na vstupu



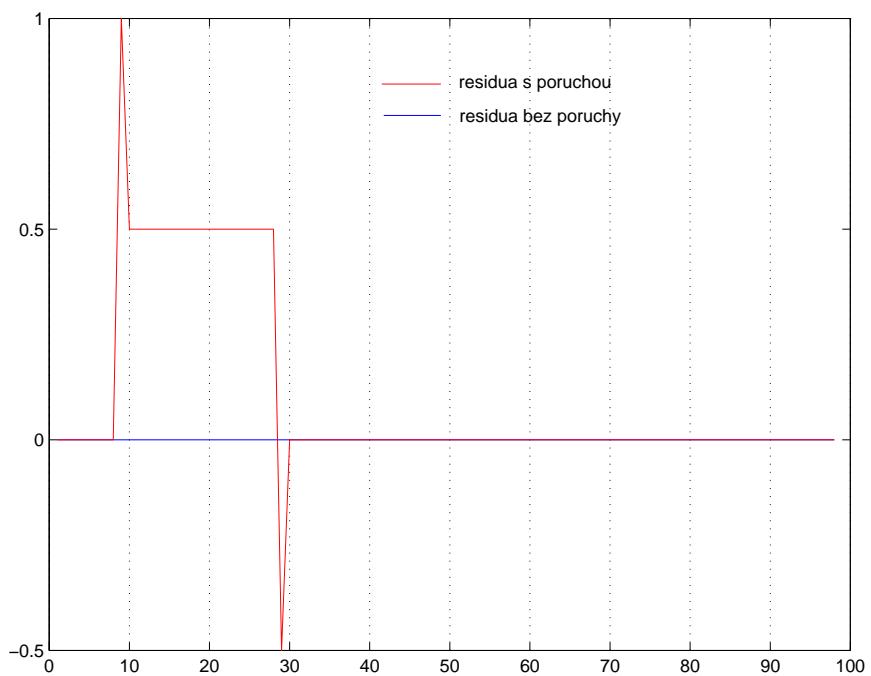
Obrázek 3.2: Residuum soustavy při poruše na vstupu



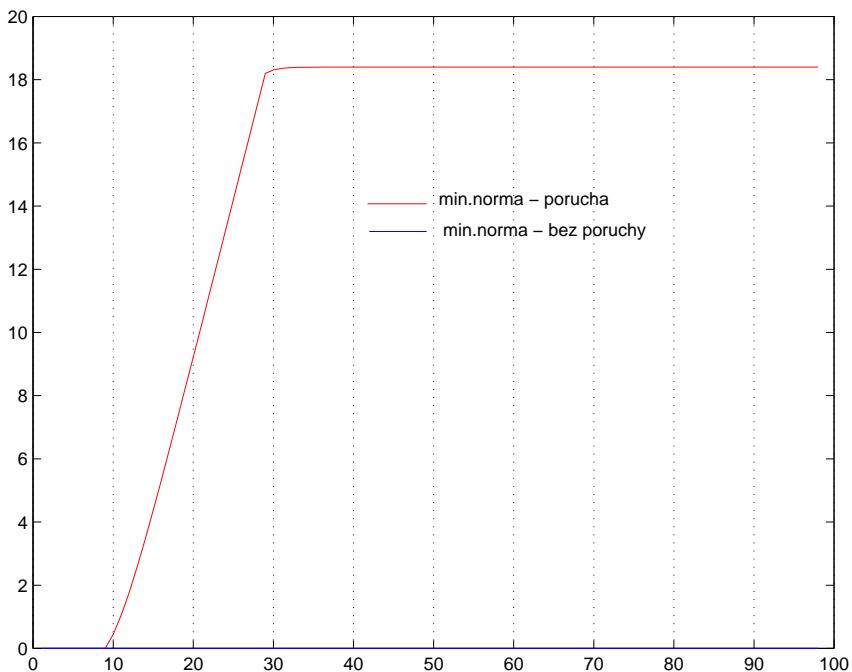
Obrázek 3.3: Minimální norma poruchového signálu z při poruše na vstupu



Obrázek 3.4: Výstup soustavy při poruše na výstupu



Obrázek 3.5: Residuum soustavy při poruše na výstupu



Obrázek 3.6: Minimální norma poruchového signálu z při poruše na výstupu

Kapitola 4

Stochastický přístup

4.1 Stochastický model s poruchami

Formálně jiný, ale principiálně stejný, přístup je stochastický pohled namísto pohledu deterministického, který reprezentoval výpočet minimální kvadratické normy poruchového signálu. Uvažujeme tedy stochastický model systému. Soustavu lineární diferenční rovnic (1.1) nahradíme soustavou lineárních diferenčních rovnic s chybovou složkou

$$\begin{aligned} y_j(t+n) + \sum_{i=0}^{n-1} a_i^j y_j(t+i) - \sum_{i=0}^n b_i^j u(t+i) = \\ d_0^j \delta(t) + \dots + d_n^j \delta(t+n) + e_j(t+n) \\ j = 1, \dots, m, \end{aligned} \tag{4.1}$$

Chybová složka e je m -dimenzionální náhodný vektor se známou hustotou pravděpodobnosti $e \sim N(0; Q_e)$, přičemž kovarianční matice Q_e je rovněž známá. Důvod, proč v rovnici (4.1) místo z píšeme δ a místo c_i píšeme d_i , vyplýne z následujících úprav. Zaved'me tedy

$$z(t) = \begin{pmatrix} \delta(t) \\ e(t) \end{pmatrix}, \quad c_i^j = \begin{cases} \begin{bmatrix} d_i^j & \vec{0} \end{bmatrix} & i \neq n \\ \begin{bmatrix} d_i^j & \vec{1}_j \end{bmatrix} & i = n \end{cases},$$

Přičemž $\vec{1}_j$ je vektor dimenze m samých nul, až na j -tou pozici, kde je 1. Potom můžeme rovnici (4.1) přepsat do tvaru, který je zcela shodný s rovnicí (dif). Proto dostáváme pro vyjádření lineárních omezení získaných měřením v čase t kompaktní zápis

$$r(t) = R \cdot z_{t+n}^t$$

A opět pro všechna omezení získaná v časovém intervalu $<0, t>$ dostaneme

$$h = r_t^0 = H \cdot z_{t+n}^0$$

a matice H je opět kvazidiagonální matice složená z matic R .

4.2 Poruchový signál jako náhodný vektor

Při stochastickém pohledu uvažujeme, že vektor z je náhodný vektor s vícedimenzionální normální hustotou pravděpodobnosti:

$$p(z) = \frac{1}{\sqrt{(2\pi)^n(z)} \cdot \sqrt{\det Q}} \exp\left\{-\frac{1}{2}z^T Q^{-1} z\right\}$$

$$Q = \begin{bmatrix} Q_\delta & O \\ O & Q_e \end{bmatrix}$$

4.3 Význam kovarianční matice pro detekci

Cílem je vypočítat pravděpodobnosti různých kovariančních maticí Q_δ podmíněné měřenými daty (rezidua r). Změnou prvků matice Q_δ (především těch na diagonále) vytváříme různé hypotézy o velikostech jednotlivých poruch. Tak například kovarianční matice $Q_\delta = \begin{pmatrix} 100 & 0 \\ 0 & 1 \end{pmatrix}$ (pro dvousložkový vektor δ) reprezentuje skutečnost, že první složka vektoru δ má velkou kovarianci (tedy rozptyl), neboli že je pravděpodobné, že první porucha je nenulová. Naopak kovarianční matice $Q_\delta = \begin{pmatrix} 1 & 0 \\ 0 & 100 \end{pmatrix}$ reprezentuje opačnou situaci, kdy je první porucha nulová a druhá nenulová.

Hlavní rozdíl oproti deterministickému přístupu je v tom, že zatímco minimální hodnota dané kvadratické formy měla pro nás sama o sobě (bez hodnot jiných kvadratických forem) informační přínos, u stochastického přístupu tomu tak není. Je to způsobeno tím, že jednotlivé pravděpodobnosti nejsou normovány a tudíž nám samy o sobě nic neříkají.

Tak například kdybychom počítali pravděpodobnost kovarianční matice, která by reprezentovala působení první poruchy, a mezi dvěma výpočty by se velikost této poruchy zvětšila, pak by pravděpodobnost, vypočtená při působení větší poruchy, byla menší než pravděpodobnost počítaná jako první. To je způsobeno tím, jak modelujeme jednotlivé poruchy. Tedy změnou kovarianční matice hustoty pravděpodobnosti, nikoliv střední hodnotou. Tudíž maximální hodnota pravděpodobnosti je v počátku (nulový vektor poruch δ) a od něj monotónně klesá. Proto naším cílem je vypočítat, kolikrát je pravděpodobnější jedna kovarianční matice oproti té druhé. Jinak řečeno

množina všech kovariančních matic, u kterých počítáme jejich pravděpodobnosti, nám tvoří množinu všech elementárních jevů. Na této množině počítáme rozložení hustoty pravděpodobnosti (samozřejmě diskrétní). To tedy v praxi znamená, že každou vypočtenou pravděpodobnost normujeme. A to tak, že ji vydělíme součtem všech vypočtených pravděpodobností. Potom nám všechny pravděpodobnosti dají v součtu hodnotu 1 (100%). Už z toho je patrné, že počítat pravděpodobnost pouze jedné kovarianční matice, je nesmyslné, neboť bychom bez ohledu na měřená data dostávali vždy stejnou hodnotu (právě 1).

4.4 Výpočet pravděpodobnosti kovarianční matice

Přejděme tedy k vlastnímu výpočtu. Počítáme tedy pravděpodobnost, že poruchový signál z má dané rozložení hustoty pravděpodobnosti (tedy gaussovské s danou kovarianční maticí Q), podmíněno pozorovanými daty, které uvažujeme ve formě soustavy lineárních rovnic $h = r_t^0 = H \cdot z_{t+n}^0$. Tuto podmíněnou pravděpodobnost v čase t můžeme vyjádřit pomocí Bayesova vztahu

$$P_{Q(i)}(t) = p(Q(i) | h) = \frac{p(h | Q(i)) \cdot p(Q(i))}{p(h)}.$$

Jelikož $p(h)$ je konstanta nezávislá na $Q(i)$ a obecně platí $\int p(x, y | \dots) dx = p(y | \dots)$, dostaneme

$$p(Q(i) | h) \propto \int p(z, h | Q(i)) dz \cdot p(Q(i)),$$

$p(Q(i))$ je apriorní rozdělení pravděpodobnosti matice Q , které uvažujeme rovnoměrné, tedy je to opět konstanta nezávislá na konkrétním $Q(i)$. Dále $p(z, h | Q(i))$ je pro $h = r_t^0 = H \cdot z_{t+n}^0$ rovna $p(z | Q(i))$ a pro ostatní h je rovna nule. Označíme-li S lineární podprostor v z , kde platí $r_t^0 = H \cdot z_{t+n}^0$ můžeme psát

$$p(Q(i) | h) \propto \int_S p(z | Q(i)) dz. \quad (4.2)$$

Přesnou hodnotu $p(Q(i) | h)$ dostaneme znormováním jednotlivých hodnot integrálů z rovnice (4.2)

$$p(Q(i) \mid h) = \frac{\int_S p(z \mid Q(i)) dz}{\sum_{i=1}^N \left(\int_S p(z \mid Q(i)) dz \right)}$$

4.5 Rekurzivní výpočet pravděpodobnosti kovarianční matice

4.5.1 Motivace

Je zřejmé, že rovnice (4.2) opět řeší náš problém pouze principiálně. Pro praktický výpočet je jen těžko použitelná, neboť bychom museli počítat integrál z n -rozměrné funkce, přičemž n by se v každém kroku zvětšovalo. Proto je nutné, abychom převedli tento výpočet do rekurzivní podoby.

4.5.2 Formální úpravy

Nejprve formálně upravíme výraz pro hustotu pravděpodobnosti $p(z(t) \mid Q)$

$$\begin{aligned} P &\equiv Q^{-1}, \det Q = \frac{1}{\det Q^{-1}} = \frac{1}{\det P} \\ p(z(t) \mid Q) &= \frac{\sqrt{\det P}}{\sqrt{(2\pi)^{n(z)}}} \exp \left\{ -\frac{1}{2} z(t)^T P z(t) \right\}. \end{aligned} \quad (4.3)$$

Nyní provedeme Choleskyho rozklad matice P , abychom mohli kvadratickou formu v exponentu hustoty (4.3) přepsat jako kvadratickou normu vektoru $(\text{chol}(P)z)$

$$\begin{aligned} P &= m^T m \\ \det P &= \det m^T \cdot \det m = (\det m)^2 \\ p(z(t) \mid Q) &= \frac{|\det m|}{\sqrt{(2\pi)^{n(z)}}} \exp \left\{ -\frac{1}{2} \|m z(t)\|^2 \right\}. \end{aligned} \quad (4.4)$$

Jelikož máme nehomogenní lineární omezení, rozšíříme vektor z o absolutní člen

$$p(z(t) \mid Q) = \frac{|\det m|}{\sqrt{(2\pi)^{n(z)}}} \exp \left\{ -\frac{1}{2} \left\| \begin{pmatrix} m & o \\ o' & 0 \end{pmatrix} \begin{pmatrix} z(t) \\ 1 \end{pmatrix} \right\|^2 \right\}. \quad (4.5)$$

4.5.3 Integrace hustoty pravděpodobnosti přes lineární podprostor

Nyní ukážeme, jak vypočítat $p(Q(i) \mid r_t^0)$, tedy pravděpodobnost $Q(i)$ podmíněnou všemi daty získanými do časového okamžiku t , za předpokladu, že známe hustotu pravděpodobnosti z podmíněnou maticí $Q(i)$ a všemi daty získanými do časového okamžiku $(t-1)$. Platí totiž

$$\int_{Hz_{t+n}^0 = r_t^0} p(z_{t+n}^0 \mid Q(i)) dz_{t+n}^0 = \int_{Rz_{t+n}^t = r(t)} p(z_{t+n}^t \mid Q(i), r_{t-1}^0) dz_{t+n}^t.$$

Budeme tedy počítat integrál z hustoty $p(z)$ přes lineární podprostor S , který je tvořen všemi řešeními soustavy lineárních rovnic $Rz = r(t)$. Jak tuto integrovanou hustotu budeme získávat, ukážeme později.

Mějme tedy v čase t hustotu pravděpodobnosti z podmíněnou danou kovarianční maticí $Q(i)$ a daty r_{t-1}^0

$$p(z_{t+n}^t \mid Q(i), r_{t-1}^0) = \frac{|\det M(t)|}{\sqrt{(2\pi)^{(n+1)n(z)}}} \exp \left\{ -\frac{1}{2} \left\| \begin{pmatrix} M(t) & m_2(t) \\ o' & 0 \end{pmatrix} \begin{pmatrix} z_{t+n}^t \\ 1 \end{pmatrix} \right\|^2 \right\} \quad (4.6)$$

a budeme počítat integrál

$$\int_{Rz_{t+n}^t = r(t)} p(z_{t+n}^t \mid Q(i), r_{t-1}^0) dz_{t+n}^t. \quad (4.7)$$

Daná omezení z času t zahrneme do (4.6) principiálně stejným způsobem, jakým jsme zahrnovali omezení při výpočtu minimální kvadratické normy v části 3.3.4. Provedeme tedy substituci $\begin{pmatrix} z_{t+n}^t \\ 1 \end{pmatrix} = A \begin{pmatrix} z_{t+n}^{t_n(r)+1} \\ 1 \end{pmatrix}$, kde matice A bude sestavená stejně jako v části (ref). Po provedení této substituce již nebude matice v jádru kvadratické normy horní trojúhelníková, a proto provedeme její QR rozklad. Rozdělíme-li matici R tohoto rozkladu na jednotlivé bloky

$$\begin{pmatrix} N(t) & n_{12}(t) \\ o' & n_{22}(t) \end{pmatrix}, \quad \text{dostaneme}$$

$$f \left(z_{t+n}^{t_n(r)+1}, Q(i), r_t^0 \right) = \frac{|\det M(t)|}{\sqrt{(2\pi)^{(n+1)n(z)}}} \exp \left\{ -\frac{1}{2} \left\| \begin{pmatrix} N(t) & n_{12}(t) \\ o' & n_{22}(t) \end{pmatrix} \begin{pmatrix} z_{t+n}^{t_n(r)+1} \\ 1 \end{pmatrix} \right\|^2 \right\} \quad (4.8)$$

Důvodem, proč místo podmíněné hustoty pravděpodobnosti $p(z_{t+n}^{t_n(r)+1} | Q(i), r_{t-1}^0)$ píšeme obecnou funkci $f(z_{t+n}^{t_n(r)+1}, Q(i), r_{t-1}^0)$, je fakt, že funkce f není hustotou pravděpodobnosti, neboť integrál z ní není roven jedné. Ale pro výpočet integrálu (4.7) platí

$$\int_{Rz_{t+n}^t = r(t)} p(z_{t+n}^t | Q(i), r_{t-1}^0) dz_{t+n}^t = \int f(z_{t+n}^{t_n(r)+1}, Q(i), r_t^0) dz_{t+n}^{t_n(r)+1} \quad (4.9)$$

K výpočtu integrálu na pravé straně rovnice (4.9) je užitečné si kvadratickou normu v exponentu hustoty pravděpodobnosti (4.8) rozdělit na součet dvou kvadratických norem, přičemž ta první bude obsahovat proměnnou z , ale ta druhá pouze absolutní člen. Provedeme tedy následující úpravy (pro zjednodušení zápisu vynecháme časové indexy)

$$\left\| \begin{pmatrix} N & n_{12} \\ o' & n_{22} \end{pmatrix} \begin{pmatrix} z \\ 1 \end{pmatrix} \right\|^2 \equiv \left\| \begin{pmatrix} F_2 & O \\ o' & f_3 \end{pmatrix} \begin{pmatrix} z - f_1 \\ 1 \end{pmatrix} \right\|^2 \quad (4.10)$$

Roznásobením levé i pravé strany ekvivalence (4.10) dostaneme

$$\left\| \begin{pmatrix} Nz + n_{12} \\ n_{22} \end{pmatrix} \right\|^2 = \left\| \begin{pmatrix} F_2 z - F_2 f_1 \\ f_3 \end{pmatrix} \right\|^2 \quad (4.11)$$

Porovnáním odpovídajících složek v (4.11) dostaneme

$$\begin{aligned} F_2 &= N \\ f_1 &= -N^{-1}n_{12} \\ f_3 &= n_{22} \end{aligned}$$

Potom můžeme ”hustotu pravděpodobnosti” (4.8) upravit

$$\begin{aligned} f(z_{t+n}^{t_n(r)+1}, Q(i), r_t^0) &= \frac{|\det M(t)|}{\sqrt{(2\pi)^{n(r)} |\det N(t)|}} \cdot \frac{|\det N(t)|}{\sqrt{(2\pi)^{(n+1)n(z)-n(r)}}} \times \\ &\times \exp \left\{ -\frac{1}{2} \left\| N(t) \left(z_{t+n}^{t_n(r)+1} + N^{-1}(t)n_{12}(t) \right) \right\|^2 \right\} \cdot \exp \left\{ -\frac{1}{2} n_{22}^2(t) \right\} = \\ &= \frac{|\det M(t)|}{\sqrt{(2\pi)^{n(r)} |\det N(t)|}} \exp \left\{ -\frac{1}{2} n_{22}^2(t) \right\} \times \\ &\quad \times \mathcal{N}(-N^{-1}(t)n_{12}(t), (N^T N)^{-1}) \quad (4.12) \end{aligned}$$

a integrál z (4.12) je roven

$$\begin{aligned}
P_{Q(i)}(t) &= \int f(z_{t+n}^{t_n(r)+1} | Q(i), r_t^0) dz_{t+n}^{t_n(r)+1} = \\
&= \frac{|\det M(t)|}{\sqrt{(2\pi)^{n(r)}} |\det N(t)|} \exp \left\{ -\frac{1}{2} n_{22}^2(t) \right\}
\end{aligned} \tag{4.13}$$

Tím jsme získali konečný výsledek v čase t , ale vycházeli jsme ze znalosti podmíněné hustoty pravděpodobnosti $p(z_{t+n}^t | Q, r_{t-1}^0)$. Proto musíme ukázat, jak tuto hustotu v získáme.

4.5.4 Rekurzivní integrace

Stejně jako v kapitole, kde jsme počítali minimální kvadratickou normu, i zde využijeme kvazidiagonalitu matice H . Tedy faktu, že data $r(t), r(t+1), \dots$ nezávisí na $z(t-1), z(t-2), \dots$. Proto můžeme provést celou integraci postupnou dříčí integrací přes jednotlivé vektory $z(t)$. Tím získáme rekurzivní předpis pro $p(z_{t+n}^t | Q, r_{t-1}^0)$

$$\begin{aligned}
p(z_{t+n}^t | Q, r_{t-1}^0) &= p(z(t+n) | Q) \\
&\quad \cdot \int_{Rz_{t+n-1}^{t-1}=r(t-1)} p(z_{t+n-1}^{t-1} | Q, r_{t-2}^0) dz(t-1) = \\
&= p(z(t+n) | Q) \int_{Rz_{t+n-1}^{t-1}=r(t-1)} p(z(t+n-1) | Q) \\
&\quad \cdot \int_{Rz_{t+n-2}^{t-2}=r(t-2)} p(z_{t+n-2}^{t-2} | Q, r_{t-3}^0) dz(t-2) dz(t-1) = \\
&= p(z(t+n)) \int_{Rz_{t+n-1}^{t-1}=r(t-1)} p(z(t+n-1) | Q) \\
&\quad \cdot \int_{Rz_{t+n-2}^{t-2}=r(t-2)} p(z(t+n-2) | Q) \\
&\quad \cdot \int \dots \int_{Rz_n^0=r(0)} p(z_n^0 | Q) dz(0) \dots dz(t-2) dz(t-1).
\end{aligned}$$

Jeden krok této rekurze ukážeme na vyjádření podmíněné hustoty pravděpodobnosti $p(z_{t+n+1}^{t+1} | Q(i), r_t^0)$. Z rovnice (4.12) získáme

$$p\left(z_{t+n+1}^{t_n(r)+1} \mid Q(i), r_t^0\right) = \frac{|\det N(t)|}{\sqrt{(2\pi)^{(n+1)n(z)-n(r)}}} \exp \left\{-\frac{1}{2} \left\|N(t)\left(z_{t+n}^{t_n(r)+1} + N^{-1}(t)n_{12}(t)\right)\right\|^2\right\}.$$

Jelikož složky vektoru $z_{t+n}^{t_n(r)+1}$ s časovým indexem t (tedy vektor $z_t^{t_n(r)+1}$, který označíme z_1) se již nemohou vyskytovat v budoucích datech, zajímá nás pouze marginální hustota pravděpodobnosti vektoru z_{t+n}^{t+1} (tento vektor označíme z_2). Analogicky jako v rovnicích (4.10) a (4.11) upravíme kvadratickou normu v exponentu (4.14). Označíme-li

$$\begin{aligned} N &= \begin{pmatrix} N_{11} & N_{12} \\ O & N_{22} \end{pmatrix}, \quad -N^{-1}(t)n_{12}(t) = \mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \\ z_{t+n}^{t_n(r)+1} &= \begin{pmatrix} z_t^{t_n(r)+1} \\ z_{t+n}^{t+1} \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}, \end{aligned}$$

pak můžeme psát

$$\begin{aligned} \left\| \begin{pmatrix} N_{11} & N_{12} \\ O & N_{22} \end{pmatrix} \begin{pmatrix} z_1 - \mu_1 \\ z_2 - \mu_2 \end{pmatrix} \right\|^2 &= \left\| \begin{pmatrix} F_1 & O \\ O & F_2 \end{pmatrix} \begin{pmatrix} (z_1 - \mu_1) - f_3 \\ (z_2 - \mu_2) \end{pmatrix} \right\|^2 \\ &\Rightarrow F_2 = N_{22}, F_1 = \dots \end{aligned} \tag{4.14}$$

Pro vyjádření μ_2 , vyjádříme nejprve inverzi matice N . Přímým výpočtem lze ověřit, že je zřejmě rovna

$$N^{-1}(t) = \begin{pmatrix} N_{11}^{-1} & -N_{11}^{-1}N_{12}N_{22}^{-1} \\ O & N_{22}^{-1} \end{pmatrix}.$$

Označíme-li $n_{12}(t) = \binom{n_{121}}{n_{122}}$, dostaneme

$$\mu = -N^{-1}(t)n_{12}(t) \Rightarrow \mu_2 = -N_{22}^{-1}n_{122}.$$

Nyní již můžeme vyjádřit hledanou hustotu pravděpodobnosti

$$p\left(z_{t+n}^{t+1} \mid Q(i), r_t^0\right) = \mathcal{N}\left(-N_{22}^{-1}n_{122}, (N_{22}^T N_{22})^{-1}\right).$$

Podmíněnou hustotu pravděpodobnosti vektoru z_{t+n+1}^{t+1} již získáme snadno

$$p(z_{t+n+1}^{t+1} | Q(i), r_t^0) = \mathcal{N}(-N_{22}^{-1}n_{122}, (N_{22}^T N_{22})^{-1}) \times \mathcal{N}(0, (m^T m)^{-1}). \quad (4.15)$$

Abychom tuto hustotu dostali do tvaru, v kterém byla uvedena hustota v (4.6), provedeme jednoduché úpravy kvadratické normy v exponentu hustoty pravděpodobnosti (4.15)

$$\begin{aligned} \|N_{22}(z_{t+n}^{t+1} - \mu_2)\|^2 + \|m z(t+n+1)\|^2 &= \left\| \begin{pmatrix} N_{22} & O \\ O & m \end{pmatrix} \begin{pmatrix} z_{t+n}^{t+1} - \mu_2 \\ z(t+n+1) \end{pmatrix} \right\|^2 \\ &= \left\| \begin{pmatrix} M(t+1) & m_2(t+1) \\ o' & 0 \end{pmatrix} \begin{pmatrix} z_{t+n+1}^{t+1} \\ 1 \end{pmatrix} \right\|^2, \end{aligned}$$

kde $M(t+1) = \begin{pmatrix} N_{22}(t) & O \\ O & m \end{pmatrix}$, $m_2(t+1) = \begin{pmatrix} -N_{22}(t)\mu_2 \\ O \end{pmatrix} = \begin{pmatrix} n_{122} \\ O \end{pmatrix}$.

4.6 Příklad

Celý algoritmus popsaný v této kapitole budeme demonstrovat na systému z kapitoly 2.1. Jak jsme popsali na začátku této kapitoly, rozšíříme popis systému o chybovou složku e (šum)

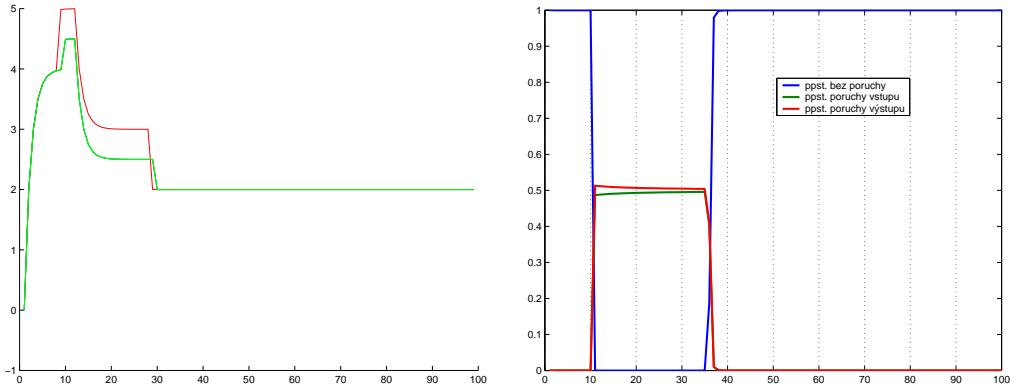
$$\begin{aligned} y(t+2) - 0.5y(t+1) - u(t+1) + u(t) &= r(t) = \\ &= [-1 \ 0] \delta(t) + [1 \ 0.5] \delta(t+1) + [0 \ -1] \delta(t+2) + e(t+2) = \\ &= [-1 \ 0 \ 0] z(t) + [1 \ 0.5 \ 0] z(t+1) + [0 \ -1 \ 1] z(t+2). \end{aligned}$$

Matice R je tedy

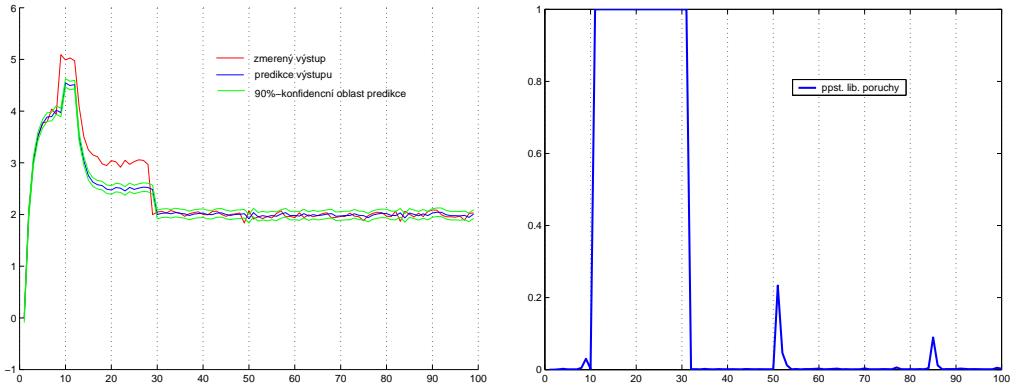
$$R = [-1 \ 0 \ 0 \ 1 \ 0.5 \ 0 \ 0 \ -1 \ 1].$$

Budeme počítat pravděpodobnosti tří kovariančních matic, odpovídajících po řadě situaci bez poruchy, s poruchou na vstupu a s poruchou na výstupu. Bude-li σ_e (tedy odmocnina z rozptylu šumu) např. 0.05, pak jednotlivé matice m (tedy choleskyho faktory příslušných informačních matic) mohou vypadat typicky takto

$$m_1 = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 20 \end{bmatrix}, m_2 = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 20 \end{bmatrix}, m_3 = \begin{bmatrix} 100 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & 20 \end{bmatrix}.$$



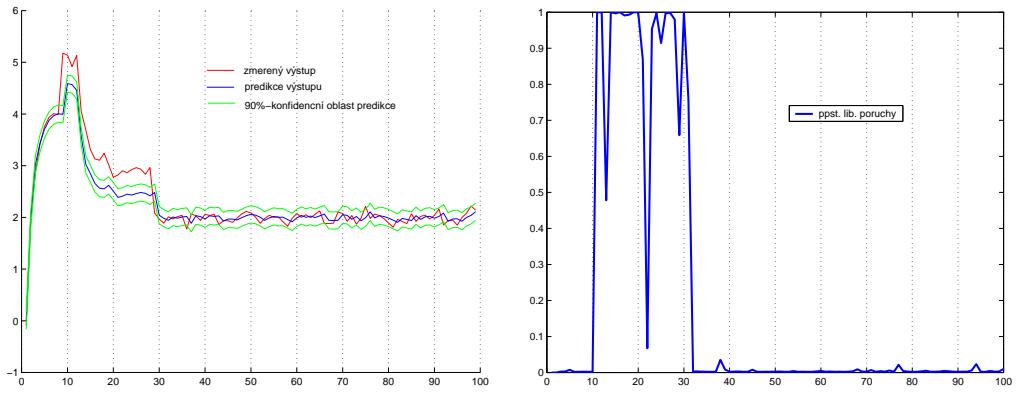
Obrázek 4.1: Predikce výstupu a pravděp. jednotlivých poruch při poruše na výstupu a $\sigma_e = 1e - 4$



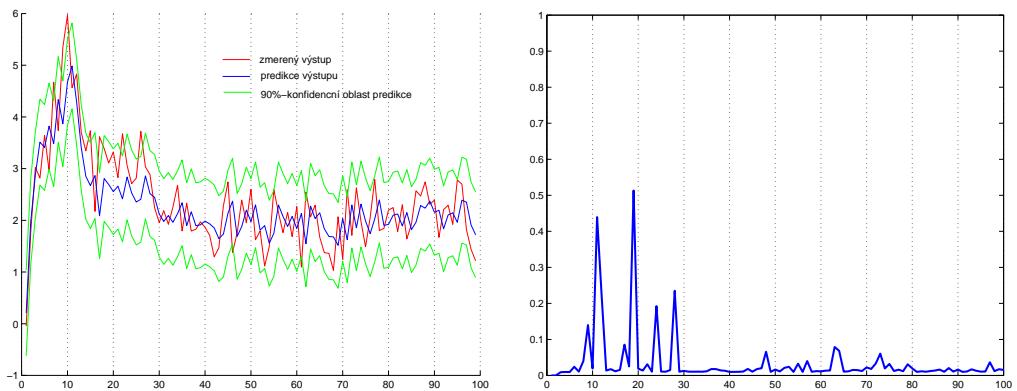
Obrázek 4.2: Predikce výstupu a pravděp. poruchy (libovolné) při poruše na výstupu a $\sigma_e = 0.05$

Na obr.4.1 až 4.4 jsou výsledky simulací pro poruchu o velikosti 1 působící na výstup soustavy od času $t = 10$ do času $t = 30$. Jednotlivé průběhy se liší velikostí rozptylu šumu působící rovněž na výstupu. Je patrné, že detekce poruchy je tím lepší, čím větší je poměr mezi velikostí poruchy a šumu (poměr signál-šum). To je obzvláště zřejmé z obr.4.4, kde velikost šumu je tak velká, že výstup zatížený poruchou leží v oblasti, kde lze s 90% pravděpodobností očekávat hodnotu výstupu. Proto také porucha detekována není.

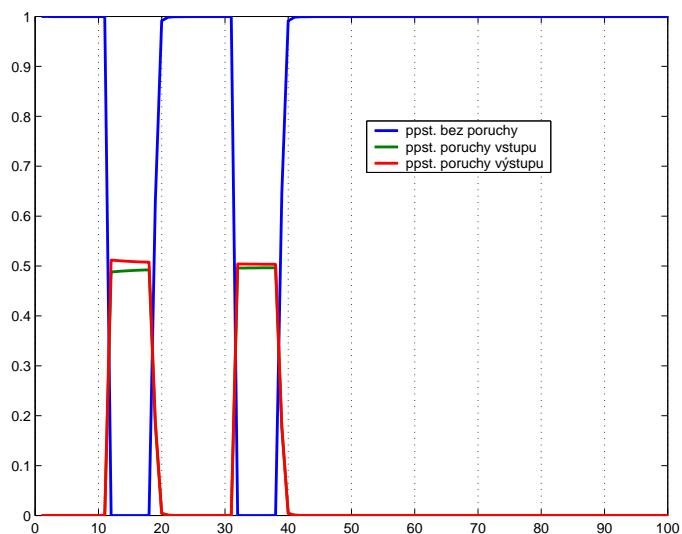
Na obr.4.5 je výsledek detekce pro poruchu působící na vstup a $\sigma_e = 1e - 4$, pro porovnání s výsledky detekce pro deterministický případ.



Obrázek 4.3: Predikce výstupu a pravděp. poruchy (libovolné) při poruše na výstupu a $\sigma_e = 0.1$



Obrázek 4.4: Predikce výstupu a pravděp. poruchy (libovolné) při poruše na výstupu a $\sigma_e = 0.5$



Obrázek 4.5: Pravděp. jednotlivých poruch při poruše na vstupu a $\sigma_e = 1e-4$

Kapitola 5

Aplikace na reálném případě z praxe - Řízení spalování uhlí v teplárně Otrokovice

5.1 Úvod

Algoritmus detekce poruch jsme aplikovali na detekci poruch analyzátorů spalin v procesu spalování uhlí v teplárně Otrokovice.

5.2 Řídící proces

Popišme nejprve celý řídící proces. Úkolem je prostřednictvím regulace množství hmotnostního toku paliva - uhelného prášku (F) a hmotnostního toku vzduchu (A) dosáhnout optimálních hodnot koncentrací kyslíku ($[O_2]$), oxidu uhelnatého ($[CO]$) a oxidů dusíku ($[NO_x]$) ve smyslu kritéria (stochastického, neboť jednotlivé koncentrace považujeme za náhodné procesy)

$$\min \sum_{k,t} \omega_k P\left(y_k(t, v(t)) > L_k\right) \quad (5.1)$$

Jedná se tedy o minimalizaci váženého (koeficienty w_k) součtu pravděpodobností, že veličina y_k je větší než její příslušná horní mez, prostřednictvím parametrů v (například poměr A/F). Veličiny y_k jsou tedy příslušné koncentrace $[O_2]$, $[CO]$, $[NO_x]$, přičemž pro stanovení dolních mezí budeme za y_k brát $-[O_2]$, $-[CO]$, $-[NO_x]$. Pro predikci pravděpodobností v (5.1) je použit adaptivní regresní model. V každém okamžiku, kdy jsou změřena data, se vypočítá pravděpodobnost koeficientů tohoto regresního modelu podmíněná

všemi změřenými daty. Z této podmíněné pravděpodobnosti se pak určí příslušné nejvěrohodnější odhad všech koeficientů. Pro úplnost pojmenujme, že jelikož se odhadují časově proměnné parametry, je pro časový krok odhadu parametrů použito tzv. zapomínání, konkrétně omezené zapomínání.

5.3 Úkol detekce poruch

Problém tohoto adaptivního řízení spočívá v časté poruchovosti senzorů (tzv. analyzátorů), které měří příslušné koncentrace, a jejichž výstupy přímo ovlivňují výsledné odhady parametrů modelu. Poruchy těchto analyzátorů se mohou nejčastěji projevovat buď ‘zamrznutím’ v nějaké krajní poloze, nebo mohou mít na výstupu stále náhodnou trajektorii, ale ta je stále méně korelována se skutečnými hodnotami. Řešení tohoto problému by měl přinést algoritmus automatické detekce poruch. Jeho úkolem je tedy rozhodnout, jestli daný (nebo pouze obecně nějaký) analyzátor je v poruše a jeho data tedy nemají být použita k adaptaci koeficientů modelu.

5.4 Struktura systému

Pro určení modelu systému, musíme nejprve určit strukturu celého modelu. To jinými slovy znamená, že musíme zjistit, na kterých veličinách závisí vývoj jednotlivých koncentrací $[O_2]$, $[CO]$, $[NO_x]$ v čase a na kterých naopak nezávisí (nebo jsou tyto závislosti statisticky nevýznamné, a proto si je můžeme dovolit zanedbat). Jelikož řešení tohoto problému není součástí této práce, uvedu zde pouze výsledky, které jsem sám převzal a které byly pak využity pro vlastní řešení problému detekce poruch.

Pro koncentraci $[O_2]$ existuje jednoduchý teoretický model (např. [8],[9]) ve formě stechiometrické bilance hoření paliva

$$[O_2](t) = \frac{0.21 A_a(t) - F_a(t)}{A_a(t)}, \quad (5.2)$$

kde A_a a F_a jsou skutečné hodnoty množství vzduchu a paliva v molech. Každou z těchto hodnot můžeme vyjádřit jako funkci hmotnostních toků příslušných veličin

$$\begin{aligned} A_a(t) &= \sum_{k=0}^{n_A} \Theta_k^A A(t-k) \\ F_a(t) &= \sum_{k=0}^{n_F} \Theta_k^F F(t-k), \end{aligned} \quad (5.3)$$

kde Θ_A , resp. Θ_F jsou příslušné koeficienty.

Po dosazení (5.3) do (5.2) a jednoduché úpravě dostaneme

$$\sum_{k=0}^{n_A} \Theta_k^A A(t-k) \left([O_2](t) - 0.21 \right) + \sum_{k=0}^{n_F} \Theta_k^F F(t-k) = 0 \simeq e_1(t), \quad (5.4)$$

kde e_1 je normálně rozdělený šum $e_1 \sim \mathcal{N}(0, \sigma_{e_1}^2)$.

Poznamenejme, že přestože rovnice (5.4) není lineární vzhledem k proměnným veličinám $[O_2]$ a A , je lineární v hledaných parametrech, a proto bude možno k jejich odhadu použít lineární regrese.

Jinou možností je použití „klasického“ lineárního modelu. Chybu takového modelu lze částečně zmenšit vhodnou nelineární transformací dat. Pro náš případ se ukázalo vhodným použít funkci logaritmus, takže model pro koncentraci $[O_2]$ bude mít strukturu

$$\sum_{k=0}^{n_A} \Theta_k^A \log A(t-k) + \sum_{k=0}^{n_O} \Theta_k^{O_2} \log \left([O_2](t-k) - 0.21 \right) + \sum_{k=0}^{n_F} \Theta_k^F F(t-k) = 0 \simeq e_1(t).$$

Kromě vzduchu A , který je do procesu hoření přiváděn řízeně a je i měřen, ovlivňuje hoření nezanedbatelným způsobem také vzduch, který vniká do hořáků nekontrolovaně díky různým netěsnostem. Tento vzduch budeme nazývat „falešný“ vzduch a budeme ho značit A_f . Pro jednoduchost budeme předpokládat, že jeho velikost je během celého hoření konstantní. Jeho konkrétní velikost můžeme odhadnout jednoduše metodou nejmenších čtverců

$$k_1 (A(t) + A_f(t)) ([O_2](t) - 0.21) + k_2 F(t) = 0,$$

kde konstanty k_1 a k_2 reprezentují převodní konstanty (zesílení) příslušných měřících členů. Jednoduchou úpravou dostaneme

$$\begin{aligned}
([O_2](t) - 0.21) A_f(t) + c \cdot F(t) + ([O_2](t) - 0.21) A(t) &= 0, \quad c = \frac{k_2}{k_1} \\
z(t) &= \begin{pmatrix} [O_2](t) - 0.21 & F(t) & ([O_2](t) - 0.21) A(t) \end{pmatrix} \\
Z &= z_T^0, \quad \text{theta} = \begin{pmatrix} A_f & c & 1 \end{pmatrix}^T \\
\text{theta} &= \arg \min_{\text{theta}} (\|Z \cdot \text{theta}\|^2)
\end{aligned}$$

Přičemž minimalizaci výrazu $\|Z \cdot \text{theta}\|^2$ provedeme efektivně opět pomocí QR -rozkladu matice Z . Dále již budeme pod označením A rozumět $A + A_f$.

Pro koncentrace $[CO]$, $[NO_x]$ již podobný jednoduchý teoretický model neexistuje, a proto je nutné určit příslušné závislosti z naměřených dat statistickým ověřováním hypotéz odpovídající příslušné závislosti. Jako výsledek bychom obdrželi

$$\begin{aligned}
\{[CO], [O_2]\} &\longrightarrow \{[CO]\} \\
\{[NO_x], A, F, [O_2], [CO]\} &\longrightarrow \{[NO_x]\}
\end{aligned} \tag{5.5}$$

Tato přiřazení vyjadřují skutečnost, že pokud zahrneme současné a minulé hodnoty veličin v levé složené závorce do modelu veličiny v pravé složené závorce, pak zmenšíme chybu predikce.

Ze vztahů (5.5) vyplývá, že pro koncentrace $[CO]$ a $[NO_x]$ vychází autoregresní modely (ARX), tedy modely uvažující závislost příslušné veličiny na svých minulých hodnotách. Přestože ARX modely dávají nejmenší chybu jednokrokové predikce, nemusí být vždy vhodné je použít. Jejich chyba predikce je totiž závislá (citlivá) na změně frekvenčního spektra příslušného signálu, k čemuž může dojít, aniž by se nějak výrazně zvětšila chyba měření daného signálu. Tato skutečnost je pro problém detekce poruch nevhodná, a proto místo modelů ARX používáme modely bez autoregresní závislosti (můžeme je značit CRX). Tyto modely jsou tím pádem necitlivé na změnu koeficientů autoregresní části, ale jsou citlivé pouze na změny ve vzájemných závislostech mezi jednotlivými veličinami. Pro modely CRX tedy dostaneme místo přiřazení (5.5), přiřazení

$$\begin{aligned}
\{[O_2]\} &\longrightarrow \{[CO]\} \\
\{A, F, [O_2], [CO]\} &\longrightarrow \{[NO_x]\}
\end{aligned} \tag{5.6}$$

Z (5.6) přímo vyplývá struktura diferenčních rovnic pro koncentrace $[CO]$ a $[NO_x]$

$$\begin{aligned} \log[CO](t) + \sum_{k=0}^{n_{O_2}} \Theta_k^{O_2}[O_2](t-k) &= e_2(t) \\ [NO_x](t) + \sum_{k=0}^{n_A} \Theta_k^A A(t-k) + \sum_{k=0}^{n_F} \Theta_k^F F(t-k) + \\ + \sum_{k=0}^{n_{O_2}} \Theta_k^{O_2}[O_2](t-k) + \sum_{k=0}^{n_{CO}} \Theta_k^{CO} \log[CO](t-k) &= e_3(t), \quad (5.7) \end{aligned}$$

kde e_2 , resp. e_3 jsou opět normálně rozdělené šumy s nulovou střední hodnotou a příslušným rozptylem $\sigma_{e_2}^2$, resp. $\sigma_{e_3}^2$. V rovnici (5.7) je místo samotné koncentrace $[CO]$ použit její logaritmus, neboť takový lineární model lépe odpovídá skutečnosti.

Ještě lepších výsledků dosáhneme provedeme-li transformaci dat $[NO_x]$, resp. $[CO]$ tak, aby takto vzniklá data (označíme je $lin[NO_x]$, resp. $lin[CO]$) lépe splňovala předpoklad lineární závislosti. Můžeme například provést kvadratickou transformaci, přičemž její koeficienty určíme tak, aby chom minimalizovali odchylku $lin[NO_x]$, resp. $lin[CO]$ od k -násobku $lin[O_2]$ ($lin[O_2] = \log([O_2] - 0.21)$ ve smyslu nejmenších čtverců.

$$\begin{aligned} lin[NO_x] &= \alpha_0 + \sum_{k=1}^2 \alpha_k [NO_x]^k = k \cdot lin[O_2] \\ &\quad \bar{\alpha}_0 + \sum_{k=1}^2 \bar{\alpha}_k [NO_x]^k + lin[O_2] = 0 \\ z(t) &= \begin{pmatrix} 1 & [NO_x](t) & [NO_x]^2(t) & lin[O_2](t) \end{pmatrix} \\ Z &= z_T^0 \\ \alpha &= arg \min_{\alpha} \left(\left\| Z \cdot \begin{pmatrix} \alpha \\ 1 \end{pmatrix} \right\|^2 \right) \end{aligned}$$

Před vlastním odhadováním parametrů našeho modelu je potřeba vyřešit problém, jak určit ty parametry, které jsou přesně nulové. Neboť díky zakrouhllovacím chybám, ale především díky approximativní volbě struktury modelu, nám žádný z parametrů nevyjde přesně nulový. Jednou možností je určit dolní a horní meze indexů v sumách v rovnicích (5.4) a (5.7), jinými slovy určit zpoždění a řád dané závislosti. K tomu byla opět použita statistická metoda testování příslušných hypotéz. Výsledkem je

- pro model $[O_2]$: $k_A = \{0, \dots, 3\}$; $k_F = \{5, \dots, 7\}$
- pro model $[CO]$: $k_{O_2} = \{1, \dots, 4\}$
- pro model $[NO_x]$: $k_A = \{2, \dots, 4\}$; $k_F = \{2, \dots, 4\}$; $k_{O_2} = \{0, \dots, 8\}$;
 $k_{CO} = \{0, \dots, 3\}$

Takto jsme vybrali nejlepší strukturu, ale pouze z takových, které mají nenulové parametry v souvislé časové posloupnosti, tedy všechny parametry s menším, resp. větším časovým indexem než je příslušný dolní, resp. horní index jsou nulové. To nám jistě zužuje výběr možných struktur. Jinou možností, která připouští nulové parametry na libovolném místě, je porovnat během samotné odhadovací procedury velikost příslušného parametru s jeho rozptylem. Pokud je odhad parametru menší než dvojnásobek jeho rozptylu, položí se přesně roven nule. Proto se musí odhadovací procedura provést dvakrát. V prvním průchodu se zjistí, které parametry jsou nulové, a teprve ve druhém se provede konečný odhad ostatních parametrů.

5.5 Parametry modelu

Nyní můžeme přistoupit k určení parametrů (lépe řečeno jejich odhadů) CRX modelu, jehož strukturu jsme popsali v předchozí části. Nejprve zavedeme označení. Označme n řád celého systému, tedy nejvyšší hodnotu horního indexu k v (5.4). Potom vektor dat (regresor) a vektor parametrů odpovídající dílčímu modelu pro jednu koncentraci je

$$g(t) = \left[A(t)([O_2](t) - 0.21), \dots, A(t-n)([O_2](t) - 0.21), \right. \\ \left. A(t), \dots, A(t-n), F(t), \dots, F(t-n), [O_2](t), \dots, [O_2](t-n), \right. \\ \left. \log[CO](t), \dots, \log[CO](t-n), [NO_x](t), \dots, [NO_x](t-n), 1 \right]^T$$

$$\Theta_i = \left[\theta_0^{AO_i}, \dots, \theta_n^{AO_i}, \theta_0^{A_i}, \dots, \theta_n^{A_i}, \theta_0^{F_i}, \dots, \theta_n^{F_i}, \right. \\ \left. \theta_0^{O_i}, \dots, \theta_n^{O_i}, \theta_0^{C_i}, \dots, \theta_n^{C_i}, \theta_0^{N_i}, \dots, \theta_n^{N_i}, a^i \right]^T \\ i = O_2, CO, NO_x \quad \Theta_i - \text{rozměr } (6(n+1)+1) \times 1 \equiv d \times 1$$

Matice všech parametrů je

$$\Theta^T = \begin{bmatrix} \Theta_{O_2}^T \\ \Theta_{CO}^T \\ \Theta_{NO_x}^T \end{bmatrix} \equiv \begin{bmatrix} \Theta_1^T \\ \Theta_2^T \\ \Theta_3^T \end{bmatrix} \quad (5.8)$$

Je zřejmé, že matice Θ je velmi řídká. Nulové prvky odpovídají koeficientům, které podle (5.4) jsou apriori nulové. Dále v každém řádku je jedna jednička, která zajišťuje jednoznačnost odhadu. Standardně se volí tam, kde je koeficient odpovídající příslušné modelované veličině v čase t , v našem případě jsou to tedy prvky $\Theta(1, 1) = \Theta(2, 4(n+1)+1) = \Theta(3, 5(n+1)+1) = 1 = \theta_0^{AO_1} = \theta_0^{C_2} = \theta_0^{N_3}$. Při tomto označení můžeme psát rovnice (5.4) a (5.7) v maticovém tvaru

$$g^T(t)\Theta_i = e_i(t), \quad i = 1, \dots, 3.$$

Máme-li data od času $t=0$ do času $t=T$, pak

$$\begin{aligned} g^T \binom{n}{T} \Theta_i &= G\Theta_i = e_i \binom{n}{T} = \vec{e}_i, \quad i = 1, \dots, 3, \\ \text{přičemž } p(\vec{e}_i) &\propto \sigma_{e_i}^{-1} \exp\left(-\frac{1}{2\sigma_{e_i}^2} \vec{e}_i^T \vec{e}_i\right). \end{aligned} \quad (5.9)$$

Potom pro Bayesovský odhad parametrů Θ_i (viz [4]) platí

$$\begin{aligned} p(\Theta_i, \sigma_{e_i} | \mathcal{D}^T) &\propto l(\Theta_i, \sigma_{e_i} | \mathcal{D}^T) \cdot p(\Theta_i, \sigma_{e_i}), \\ l(\Theta_i, \sigma_{e_i} | \mathcal{D}^T) &\propto \sigma_{e_i}^{-K} \exp\left(-\frac{1}{2\sigma_{e_i}^2} \Theta_i^T G^T G \Theta_i\right), \\ \text{přičemž } K &= T - n + 1, \text{ což je počet využitelných dat.} \end{aligned}$$

Neinformativní apriorní hustota pravděpodobnosti je

$$p(\Theta_i, \sigma_{e_i}) \propto \sigma_{e_i}^{-1},$$

potom

$$p(\Theta_i, \sigma_{e_i} | \mathcal{D}^T) \propto \sigma_{e_i}^{-K+1} \exp\left(-\frac{1}{2\sigma_{e_i}^2} \|\Theta_i G\|^2\right), \quad (5.10)$$

označíme-li

$$\hat{\Theta}_i = \arg\left(\min_{\Theta_i} \|\Theta_i G\|^2\right) \text{ a } f^2 = \min_{\Theta_i} \|\Theta_i G\|^2 = \left\| \hat{\Theta}_i G \right\|^2,$$

můžeme psát

$$p(\Theta_i, \sigma_{e_i} | \mathcal{D}^T) \propto \sigma_{e_i}^{-K+1} \exp\left(-\frac{f^2}{2\sigma_{e_i}^2} - \frac{1}{2\sigma_{e_i}^2} \|\Theta_i - \hat{\Theta}_i\|_G^2\right),$$

potom pro marginální hustotu pravděpodobnosti σ_{e_i} , resp. Θ_i platí

$$p(\sigma_{e_i} | \mathcal{D}^T) \sim \chi_{K-n}^2\left(\frac{f^2}{\sigma_{e_i}^2}\right) \Rightarrow \hat{\sigma}_{e_i}^2 = \mathcal{E}\{\sigma_{e_i}^2 | \mathcal{D}^T\} = \frac{f^2}{K-n},$$

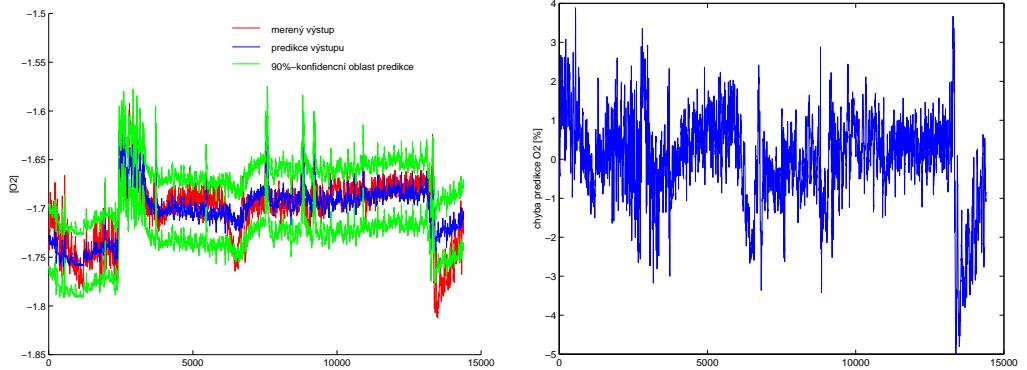
resp.

$$p(\Theta_i | \mathcal{D}^T) \sim t_n\left(\hat{\Theta}_i, \frac{f^2}{K-n}(G^T G)^{-1}, (K-n)\right),$$

kde χ_ν^2 je rozdělení chí-kvadrát s ν stupni volnosti a $t_n(EX, varX, \nu)$ je n-rozměrné Studentovo rozdělení s danou střední hodnotou a rozptylem a s ν stupni volnosti.

Vraťme se ke kvadratické formě $\|\Theta_i G\|^2$, která se vyskytuje ve výrazech pro podmíněné hustoty pravděpodobnosti v rovnici (5.10), a ukažme, jak lze efektivně numericky vypočítat hodnoty $\hat{\Theta}_i$ a f^2 , které potřebujeme pro výpočet odhadů parametrů a rozptylu šumu. Na první pohled by se mohlo zdát, že minimální hodnota této normy je nulová, a to volbou nulového vektoru Θ_i . Problém je ovšem v tom, že, jak jsme dříve poznamenali, vektor Θ_i nemůže nabývat libovolných hodnot. Některé složky jsou nulové a právě jedna jeho složka je fixována na hodnotu 1. Proto naším úkolem je řešit problém minimalizace kvadratické formy za lineárních omezení, které přímo předepisují některým složkám vektoru Θ_i pevné hodnoty. Jelikož je celý postup obdobný postupu, který jsme podrobně popsali v části 3.3, popíšeme ho pouze stručně.

Nejprve provedeme QR rozklad matice G . Předtím ale odstraníme z matice G ty sloupce, které odpovídají složkám vektoru Θ_i , které jsou apriori nulové. A dále prohodíme sloupce tak, že sloupec odpovídající složce vektoru Θ_i , která musí nabývat hodnoty jedna, bude vpravo od všech ostatních sloupců. Tako modifikovanou matici rozložíme QR rozkladem a matici R tohoto rozkladu označíme jako G_r . Matice G_r je čtvercová matice rozměru (l_i, l_i) , kde l_i je počet (apriori) nenulových složek vektoru Θ_i . Jelikož upravená matice G má mnohem více řádků (jejich počet je K - počet dat) než sloupců (těch je také l_i), můžeme provést tzv. ekonomický QR rozklad, v kterém se počítá pouze prvních l_i sloupců matice Q . Pokud je matice G_r singulární, to jest, pokud má někde na diagonále nulový prvek, upravíme tuto matici tak, aby odpovídala zobecněnému choleskyho faktoru matice $G_r^T G_r$. Tuto úpravu jsme popsali v části 3.3.5. Pro jednoduchost označme takto upravenou matici



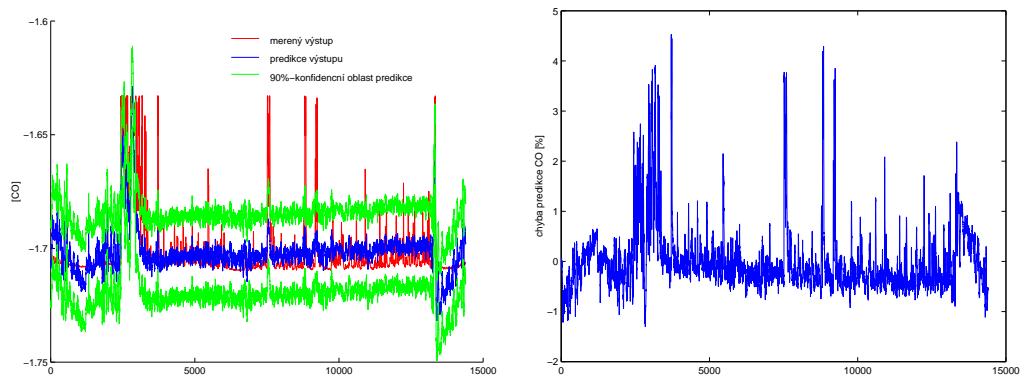
Obrázek 5.1: Predikce $[O_2]$ a chyba této predikce

opět G_r . Nyní tedy můžeme spočítat $\hat{\Theta}_i$ tak, že postupně počítáme jednotlivé složky tohoto vektoru odspodu (samozřejmě kromě poslední složky, která má hodnotu jedna), přičemž pokud narazíme na nulový prvek na diagonále, můžeme příslušnou složku vektoru Θ_i libovolně zvolit. Minimální hodnota výrazu $\|\Theta_i G\|^2$ (v rovnici (5.10) jsme ji označili f^2) je pak rovna druhé mocnině posledního prvku na diagonále matice G_r (jak jsme rovněž ukázali v části 3.3.3).

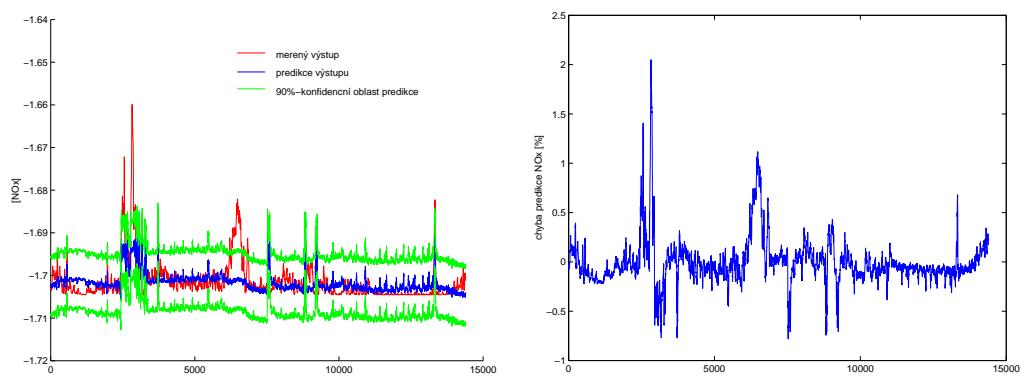
5.6 Vlastní detekce poruch

Pro úspěšnou detekci poruch je samozřejmě důležité, aby náš model odpovídal co nejlépe skutečnosti, tedy aby predikce budoucích hodnot byla blízká skutečným hodnotám. Na obr.5.1 až 5.3 jsou zobrazeny průběhy predikce pro jednotlivé koncentrace $[O_2]$, $[CO]$, $[NO_x]$ včetně relativních chyb v procentech. Predikce a následné porovnání se skutečnými hodnotami bylo provedeno na souboru dat pořízených v průběhu jednoho dne na jednom z kotlů v teplárně Otrokovice. Je vidět, že maximální chyba predikce dosahuje hodnot v řádech jednotek procent, což je vzhledem k neurčitostem v procesu spalování uspokojivé. V obrázcích, kde jsou vykresleny predikce budoucích hodnot, je rovněž vynáčena tzv. konfidenční oblast pro pravděpodobnost 90%. Tato oblast vyznačuje pro každý okamžik rozsah hodnot, v kterých leží s pravděpodobností 90% skutečná hodnota příslušné koncentrace.

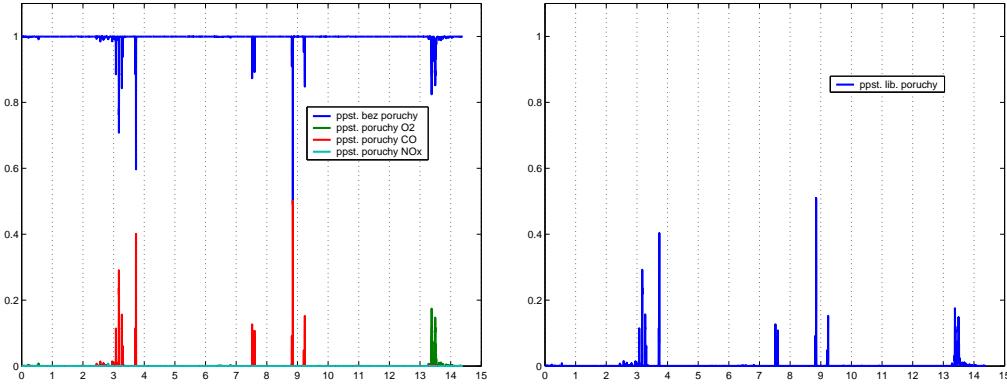
Vedle správné detekce poruch je neméně důležité nedetektovat poruchu za bezporuchového stavu. Na obr.5.4 je výsledek detekce na datech bez poruchy. První obrázek znázorňuje pravděpodobnosti jednotlivých poruch, druhý pak pravděpodobnost libovolné (z námi uvažovaných) poruchy. Je vidět, že porucha nebyla skutečně detekována až na krátkodobé impulsní ”poruchy”



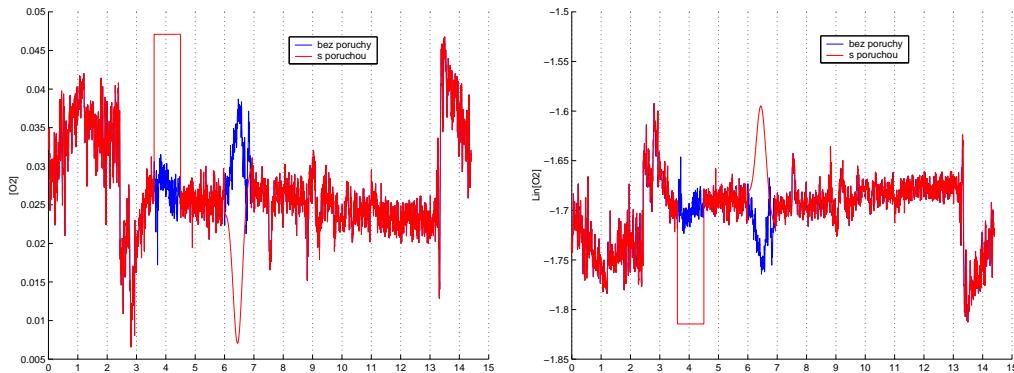
Obrázek 5.2: Predikce $[CO]$ a chyba této predikce



Obrázek 5.3: Predikce $[NO_x]$ a chyba této predikce



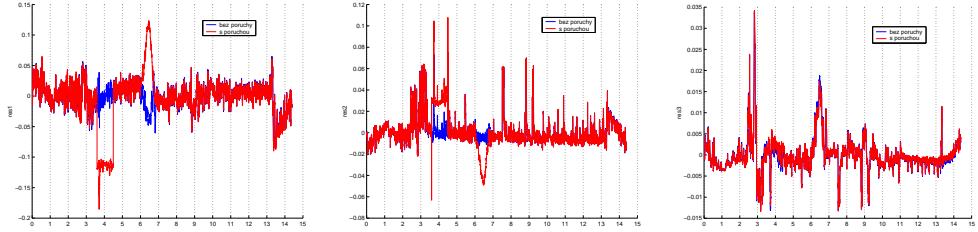
Obrázek 5.4: Pravděpodobnosti jednotlivých poruch a pravděpodobnost libovolné poruchy



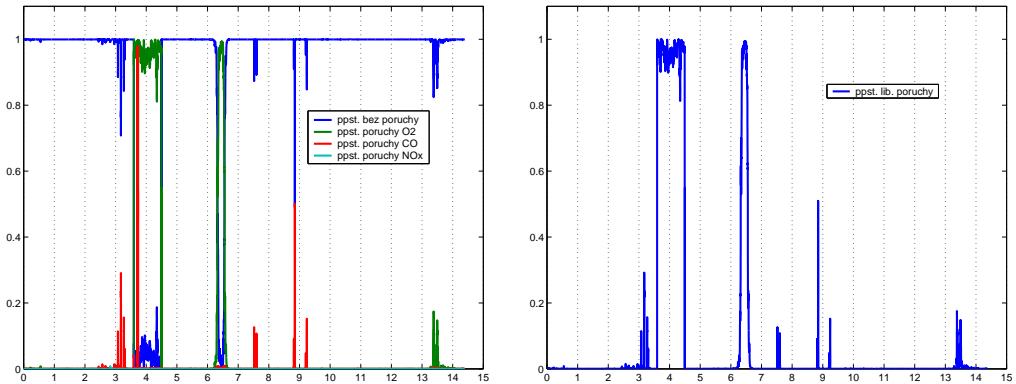
Obrázek 5.5: Poruchy $[O_2]$ v původních a transformovaných datech

$[CO]$, které nebylo možno naším modelem postihnout. I přesto tyto impulsní "poruchy" nezpůsobily nárůst pravděpodobnosti nějaké poruchy nad 50%.

Jako první uvedeme výsledky detekce při simulaci poruch na analyzátoru $[O_2]$. Uvažovali jsme dva typy poruch. Jednak skokovou poruchu s amplitudou 0.02, která působila na vzorky z intervalu $< 3600, 4500 >$, a jednak poruchu, která se postupně zvětšuje až po svoje maximum (-0.02), a pak se opět zmenšuje. Ta působila v intervalu $< 6000, 6900 >$. Obě poruchy jsou dobře patrné z obr.5.5. Na obr.5.5 je znázorněno, jak se poruchy projeví v transformovaných (linearizovaných) datech, z kterých jsou pak poruchy skutečně detekovány. Je vidět, že amplituda poruch je relativně menší vůči bezporuchové hodnotě $Lin[O_2]$ než je tomu u netransformovaných dat. To je samozřejmě nežádoucí, ale je to daň za lepší model s menší chybou predikce, a tedy za správnou detekci při bezporuchovém stavu.



Obrázek 5.6: Residua při poruchách analyzátoru $[O_2]$

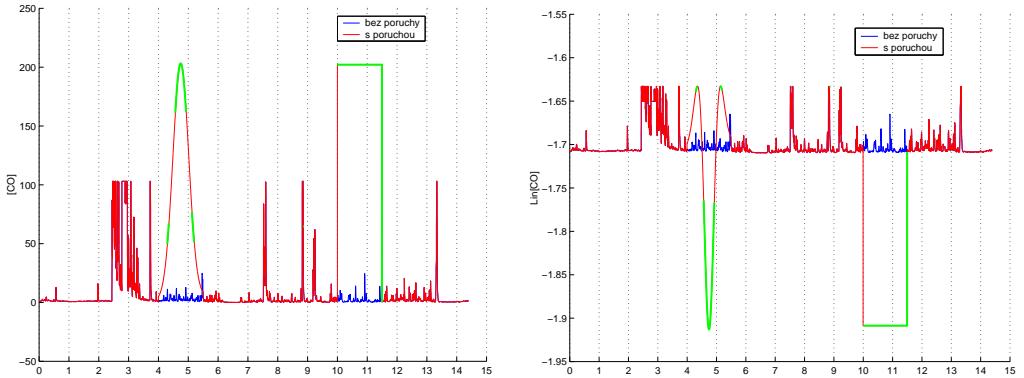


Obrázek 5.7: Pravděpodobnosti jednotlivých poruch a pravděpodobnost libovolné poruchy při poruše $[O_2]$

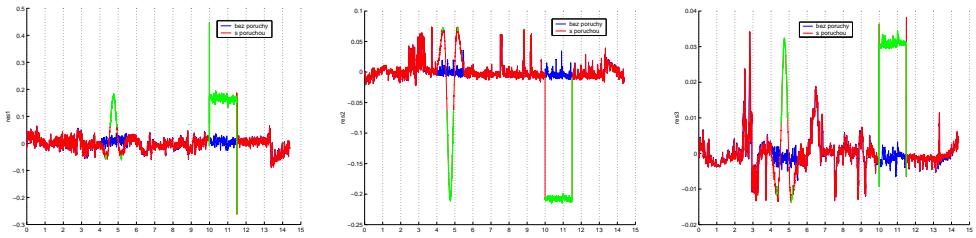
Na obr.5.6 jsou znázorněny průběhy residuů všech tří rovnic. Jak jsme již uvedli v předchozích kapitolách, residua jsou klíčová pro vlastní detekci poruch. Pro úspěšnou detekci je potřeba, aby se porucha co nejvíce v residuích projevila, což je v tomto případě splněno.

A konečně na obr.5.7 je výsledek celé detekce pro poruchy na analyzátoru $[O_2]$.

Na obr.5.8 až 5.10 jsou zobrazeny stejné průběhy jako na předchozích obrázcích, ale pro poruchy na analyzátoru $[CO]$. A to konkrétně pro poruchu skokovou s amplitudou 200 na intervalu $< 10000, 11500 >$ a pro poruchu postupně se zvětšující s amplitudou 200 na intervalu $< 4000, 5500 >$. Na obr.5.8 je dobře vidět negativní efekt, který s sebou přináší transformace dat za účelem linearizace příslušné závislosti. Zvláště u gradující poruchy je vidět, že porucha s určitou velikostí se v transformovaných datech neprojeví, a proto nemůže být detekována. Zelenou barvou je v obrázcích znázorněno, kdy pravděpodobnost poruchy (jakékoliv) je větší než padesát procent. Detekce poruch analyzátoru $[CO]$ dopadla ze všech tří detekcí nejhůře, přesněji řečeno poměr mezi poruchovým signálem a skutečným signálem musí být



Obrázek 5.8: Poruchy $[CO]$ v původních a transformovaných datech

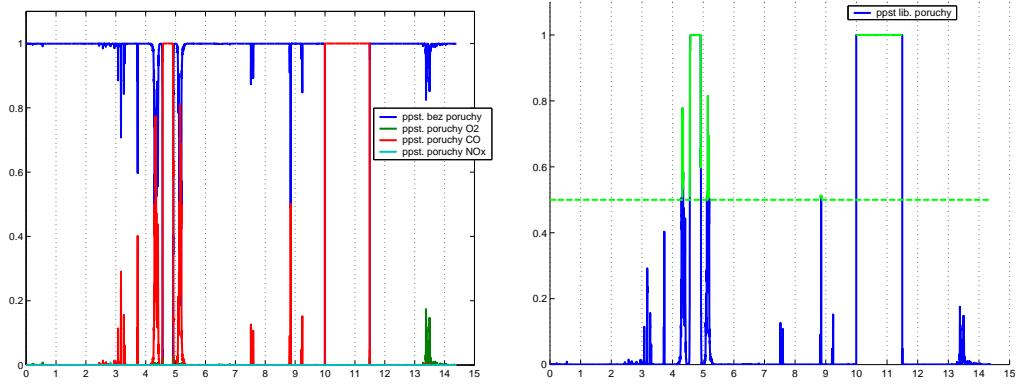


Obrázek 5.9: Residua při poruchách analyzátoru $[CO]$

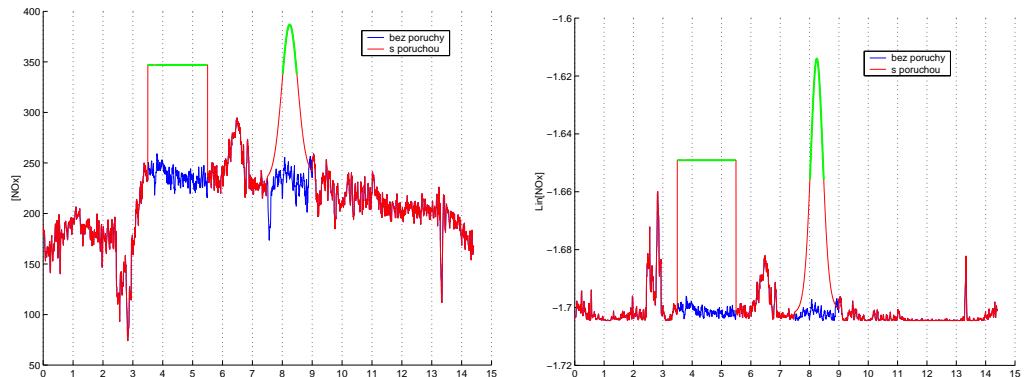
u analyzátoru $[CO]$ největší, aby byla detekována porucha.

A konečně na obr.5.11 až 5.13 jsou výsledky detekce poruch analyzátorů při simulované poruše na analyzátoru $[NO_x]$. Opět byla simulována jednak porucha skoková s amplitudou 110 na intervalu $< 3500, 5500 >$ a jednak porucha postupně se zvětšující s amplitudou 150 na intervalu $< 7500, 9000 >$.

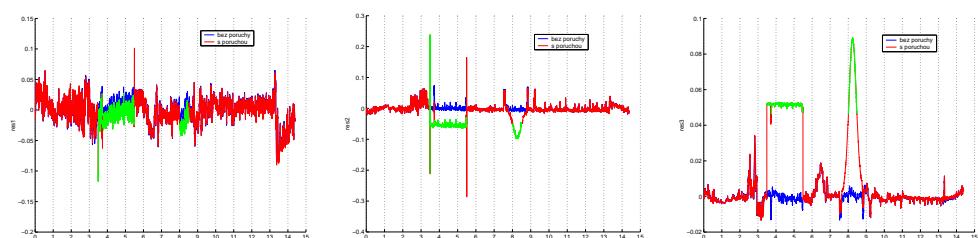
V kapitole, kde jsme se zabývali volbou struktury našeho modelu, jsme teoreticky zdůvodnili oprávněnost volby CRX modelu, tedy modelu, který neobsahuje závislost modelované veličiny na svých minulých hodnotách, ale pouze na současných a minulých hodnotách ostatních veličin. Nyní oprávněnost této volby potvrďme i experimentálně. Na obr.5.14 jsou znázorněny relativní chyby predikce (v procentech) ARX modelu pro koncentrace $[O_2]$, $[CO]$, $[NO_x]$, tak jak jsou na obr.5.1 až 5.3 znázorněny tytéž chyby predikce CRX modelu. Ze vzájemného porovnání je zřejmé, že ARX modely mají výrazně menší chybu predikce, někde i o řád. Proto také výsledek detekce poruch za bezporuchové situace bude pro tyto modely příznivější, jak dokládá obr.5.15. Naopak pro detekci poruch při výskytu nějaké poruchy, je klíčové, jak se porucha v datech projeví v residuích. Na obr.5.16 jsou zobrazena residua při poruše analyzátoru $[O_2]$. Z nich je patrné, že přenos z poruchy na residua má derivační (diferenční) charakter, a proto se konstantní porucha projeví



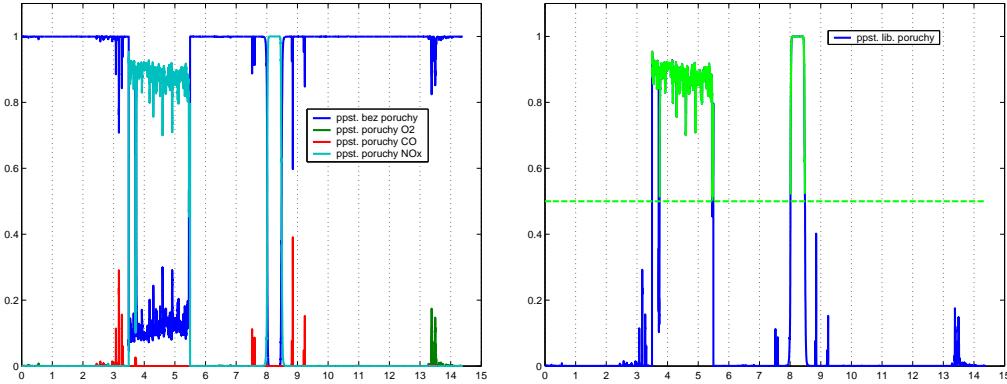
Obrázek 5.10: Pravděpodobnosti jednotlivých poruch a pravděpodobnost libovolné poruše [CO]



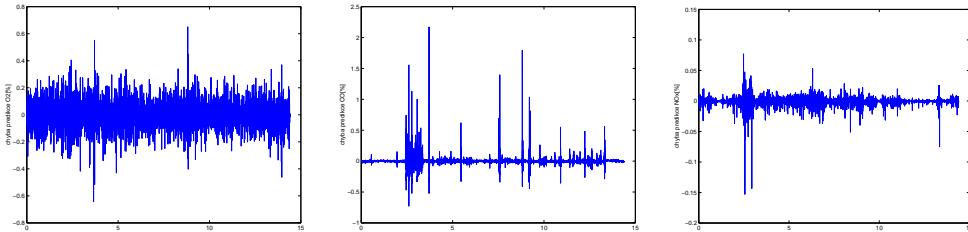
Obrázek 5.11: Poruchy [NO_x] v původních a transformovaných datech



Obrázek 5.12: Residua při poruchách analyzátoru [NO_x]



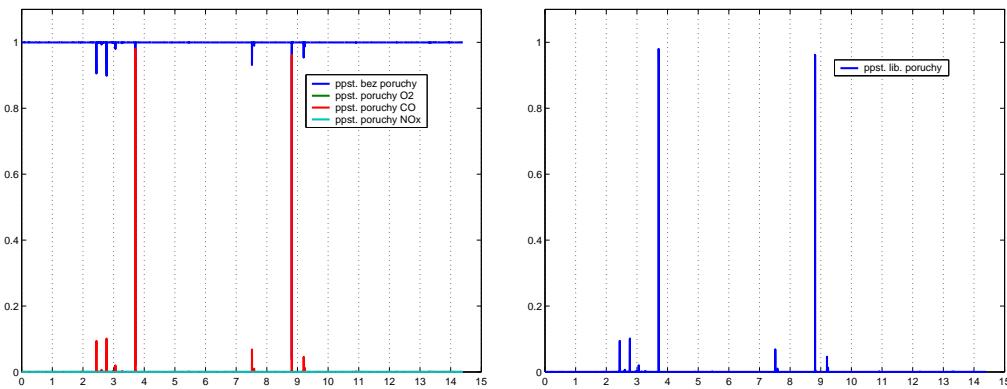
Obrázek 5.13: Pravděpodobnosti jednotlivých poruch a pravděpodobnost libovolné poruchy při poruše $[NO_x]$



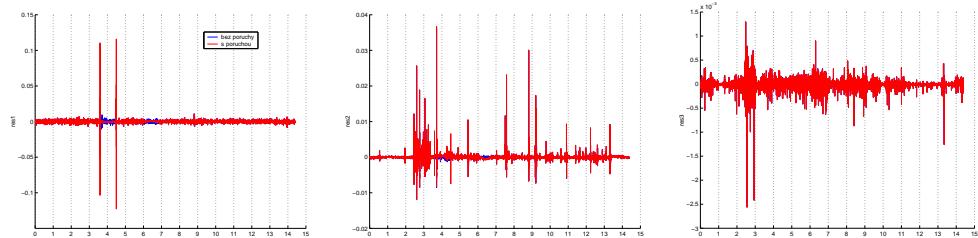
Obrázek 5.14: Relativní chyby predikce ARX modelu pro jednotlivé analyzátory

v residuech pouze při vzniku a zániku poruchy. To potvrzují i vlastní výsledky celé detekce na obr.5.17.

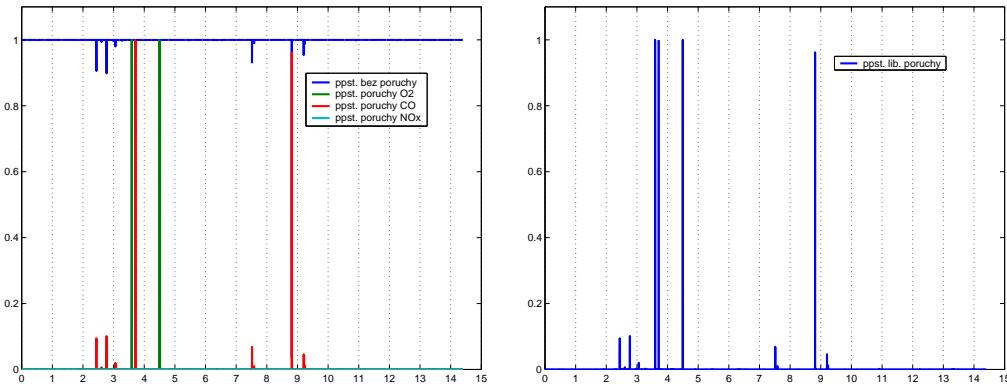
Jak jsme poznali výše, přináší transformace dat za účelem zlinearizování modelovaných závislostí pro detekci poruch i svá negativa. Především u poruch analyzátoru $[CO]$ jsme poznali, že transformace dat může znemožnit detekci poruchy o určité velikosti. Přesto je tato transformace nezbytná, jak ukazuje obr.5.18, kde je zobrazen výsledek detekce při simulované poruše analyzátoru $[CO]$, přičemž ale model byl vytvořen mezi netransformovanými veličinami. Je vidět, že chyba takového modelu je natolik velká, že dochází k detekci poruchy i za bezporuchového stavu.



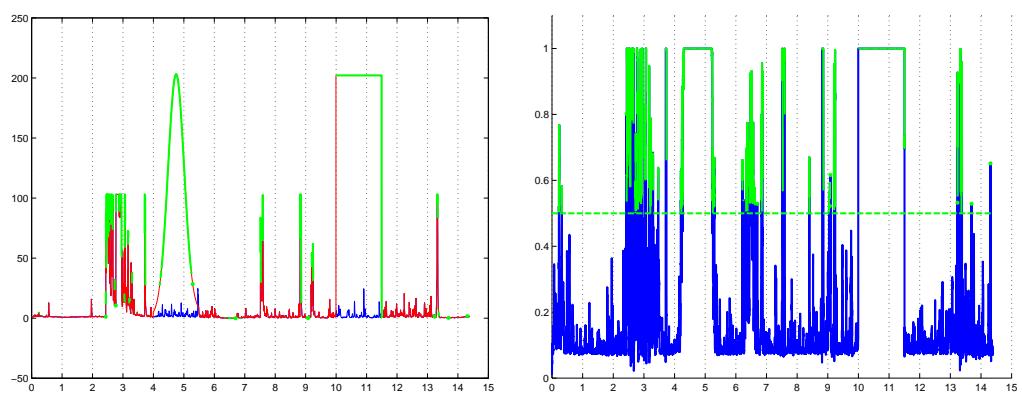
Obrázek 5.15: Pravděpodobnosti jednotlivých poruch a pravděpodobnost libovolné poruchy za bezporuchové situace



Obrázek 5.16: Residua ARX modelu při poruše analyzátoru $[O_2]$



Obrázek 5.17: Pravděpodobnosti jednotlivých poruch a pravděpodobnost libovolné poruchy při poruše $[O_2]$



Obrázek 5.18: Data $[CO]$ i s poruchami a pravděpodobnost libovolné poruchy

Závěr

Cílem této diplomové práce bylo popsat efektivní algoritmy automatické detekce poruch, a to konkrétně algoritmus výpočtu minimální normy poruchových signálů daného dynamického systému pro deterministický případ, resp. algoritmus výpočtu pravděpodobnosti různých kovariančních matic (reprezentující různé poruchy) těchto poruchových signálů (uvažovaných jako náhodné veličiny) pro stochastický případ. Zároveň bylo cílem ukázat funkčnost nejen těchto algoritmů, ale i zvoleného přístupu k detekci jako takového, na konkrétním příkladě, včetně komplexního problému z praxe.

Funkčnost obou algoritmů jsme skutečně ověřili na jednoduchém ad hoc zvoleném příkladě. Ale i tento jednoduchý příklad nám poodhalil klíčové skutečnosti, které musí být pro úspěšnou detekci dané poruchy splněny. Je to především skutečnost vyplývající z evidentního faktu, že pokud chceme z měřených dat (výstupy daného systému) detekovat danou poruchu, je nezbytné, aby se tato porucha v těchto datech co nejvíce projevila. Setkali jsme se v zásadě se dvěma obecnějšími případy, kdy toto nemusí být splněno. K prvnímu případu dojde v momentě, kdy velikost neurčitosti v měřených datech je srovnatelná nebo dokonce větší než velikost detekované poruchy. Tato neurčitost může být způsobena nepřesností (což je jiný pojem pro neurčitost) modelu nebo nepřesností měření (velký šum měření). To jsme názorně demonstrovali na stochastickém příkladě v části 4.6, kde při postupném zvětšování rozptylu šumu měření jsme dosáhli momentu, kdy již porucha nebyla detekována.

Druhý případ, kdy se porucha dostatečně neprojeví v měřených datech, je zřejmě ještě evidentnější. Dochází k němu v momentě, kdy přenos mezi uvažovanou poruchou a měřenými daty (výstupy) má takový charakter, že potlačuje nějakou významnou složku této poruchy. Na to jsme narazili již v našem zvoleném příkladě (a to jak v deterministickém případě v části 3.4, tak i v stochastickém případě v části 4.6), kde přenos mezi poruchou vstupu a měřeným výstupem měl derivační charakter, který potlačoval stejnosměrnou složku této poruchy. Na tento problém jsme pak narazili především při volbě struktury modelu pro koncentrace spalin při hoření uhlí v teplárně

Otrokovice. Zde jsme museli upustit od modelu, který dával nejmenší chybu predikce (ARX model), neboť při jeho použití docházelo k výše zmíněnému efektu, jak jsme ukázali v části 5.6. Principiálně podobný problém jsme řešili při transformaci měřených dat za účelem zlinearizování jejich vzájemné závislosti. Použití této transformace totiž na jednu stranu zmenšovalo chybu následně vytvořeného modelu, na druhou stranu ale v některých případech zmenšovalo poměr mezi poruchovým a bezporuchovým signálem v nových transformovaných datech. Zde se ale nakonec ukázalo výhodnější data transformovat, jak je patrné v části 5.6. Obecně by se dalo shrnout, že je při vytváření modelu vždy nutné volit mezi kroky, které na jedné straně vedou k zmenšení chyby výsledného modelu a tím zlepšují výsledky detekce za bezporuchového stavu (zvyšují tzv. robustnost samotné detekce), ale na druhou stranu často vedou k tomu, že se poruchy dostatečně neprojeví v měřených datech (a potažmo v residuech, které jsou pro detekci klíčové) a tím se zhorsuje schopnost detekovat poruchu.

Závěrem tedy můžeme shrnout, že algoritmy automatické detekce poruch, kterými jsme se v této diplomové práci zabývali, jsou z hlediska teoretických principů detekce poruch funkční. Jejich funkčnost v praktickém problému závisí především na tom, jak se nám skutečné poruchy, které chceme detektovat, podaří namodelovat pomocí lineárních poruchových signálů a jak tyto poruchové signály včleníme do samotného modelu daného dynamického systému.

Literatura

- [1] Basseville,M., Benveniste, A.:*Detection of abrupt changes in signals and dynamical systems.*,Lecture Notes in Control and Information Sciences,Vol.77,Springer-Verlag,Berlin,1986.
- [2] Rao,C.Radrakrishna:*Lineární metody statické indukce a její aplikace.*,Academia,Praha,1978.
- [3] Matějka,J., Zítek,P.:*Detekce poruchových stavů s využitím anichronních estimátorů.*,Dizertační práce,ČVUT,Praha,1996.
- [4] Havlena,V., Štecha,J.:*Moderní teorie řízení.*, Skriptum ČVUT,Vydavatelství ČVUT,Praha,2000.
- [5] Štecha,J., Havlena,V.:*Optimální rozhodování a řízení.*, Skriptum ČVUT,Vydavatelství ČVUT,Praha.
- [6] Himmelblau,David M.:*Fault detection and diagnosis in chemical and petrochemical processes.*, Elsevier,Amsterdam,1978.
- [7] Turkey,Adnan A.:*Fault detection and diagnosis in dynamic systems-a survey.*, Computer and Automation Institut.,Budapest,1989.
- [8] Desideri,J.A.:*Experimentation,modelling and computation in flow,turbulence and combustion..*, Vol.1,Wiley,Chichester,1995.
- [9] Warnatz,J., Maas,U., Dibble,Robert W.:*Combustion:physical and chemical fundamentals,modelling and simulation,experiments,pollutant formation.*,Springer-Verlag,Berlin,1996.
- [10] Spalding, Dudley B.:*Combustion and mass transfer:A textbook with multiple-choice exercises for engineering students.*,Springer-Verlag,Berlin,1996.
- [11] Rogalewicz,V.:*Pravděpodobnost a statistika pro inženýry..*, Skriptum ČVUT,Vydavatelství ČVUT,Praha,1998.