

## PŘEDMLUVA

Balík programů MODES slouží pro analýzu dynamického chování složité elektrizační soustavy (ES)<sup>1</sup>. Jako metoda se používá simulace přechodných dějů na matematickém modelu ES. Při popisu matematického modelu je použit objektově orientovaný přístup, kdy jednotlivé prvky ES jsou považovány za objekty.

Vstupní data pro vlastní program dynamické simulace (jmenuje se stejně jako celý balík MODES<sup>2</sup>) jsou v textovém tvaru a mají formu formuláře, tabulky, databáze a katalogu<sup>3</sup>. Ve formulářích zadává uživatel požadavky na fungování programu a s modelovanou soustavou přímo nesouvisí. Pokud uživatel pracuje pomocí uživatelského rozhraní MODMAN, nepříjde s nimi bezprostředně do styku, neboť jsou upravovány dialogovým způsobem. Základní informace o modelované ES jsou uloženy ve třech vstupních souborech, které mají standardně jména:

- VET.DAT s parametry větví (jako větve jsou modelovány vedení, trať a spínače)
- UST.DAT s parametry uzlů (jako uzly jsou modelovány přípojnice a rozvodny)
- GEN.DAT s parametry bloků (jako bloky jsou modelovány synchronní a asynchronní stroje).

Je možné používat tzv. variace chodů sítě, které se tvoří následovně. každý chod je identifikován uživatelem zadaným povinně pětímístným řetězcem &&&&& (kde za & lze dosadit libovolný alfanumerický znak bez diakritiky případně pomlčku nebo podtržítka), který musí být v rámci skladiště unikátní. Jednotlivé soubory chodu jsou pak pojmenovány:

&&&&&N.ZT.UST pro tabulku uzlů obsahující úsek uzlů, oblastí a profilů

&&&&&\_T.VET pro tabulku větví obsahující úsek větví, regulační trať

&&&&&N\_.GEN pro tabulku bloků obsahující úsek synchronních generátorů a asynchronních motorů, kde za NZT se dosadí číslo variace nasazení bloků, zatížení uzlů a topologie sítě.

Uvedené tři vstupní soubory obsahují primární (nezbytné a nenahraditelné) informace o konfiguraci sítě, distribuci zatížení do jednotlivých uzlů a nasazení jednotlivých zdrojů pro pokrytí zatížení, ztrát v síti a salda předávaných výkonů do sousedních systémů. V souboru UST.DAT jsou zároveň uloženy sekundární (vypočítané) informace o ustáleného stavu. Význam těchto souborů plyne z toho, že si je uživatel musí pořídit sám na rozdíl od databází a katalogů, které mohou být součástí dodávky (případně je může program nahradit „standardním“ způsobem implicitními hodnotami).

Výše uvedenému rozdělení vstupních souborů se vymyká vstupní soubor SCENAR.DAT, v němž je uvedena časová posloupnost zásahů, které simulují různé druhy změn a poruch v ES. Vzhledem k tomu soubor hraje klíčovou roli při vlastní dynamické simulaci, neboť při bezchybné inicializaci<sup>4</sup> a bez zadání zásahů by program MODES měl vypočítat stále průběhy (pokud systém jako takový je staticky stabilní).

Vzhledem k důležitosti těchto souborů balík MODES obsahuje tzv. pomocné programy, které usnadňují tvorbu a údržbu uvedených vstupních souborů (tzv. „pre-processing“). Jsou to tyto programy:

- PSSEData- převod dat ve formátu PSS/E (Raw data) do souborů chodu sítě (vytvoří se základní variace)
- UST - výpočet nového ustáleného stavu po změně konfigurace sítě, zatížení nebo nasazení zdrojů
- EKVIVAL - zjednodušení složité ES náhradou její části ekvivalenty
- ZKRATY
- SCENARE - tvorba souboru SCENAR.DAT, který obsahuje tzv. hromadné<sup>5</sup> scénáře vybraných zásahů
- SABLONY.

Program MODES provádí výpočet časových průběhů proměnných - tzv. dynamickou simulaci (vybrané proměnné lze zobrazit na obrazovce během výpočtu). Kromě toho provádí:

- ♦ rozbor primární a sekundární regulace P a f (uložené ve výst.souborech PRIM\_REG a SEK\_REG.ANA)
- ♦ analýzu stavu sítě na začátku a konci simulace (uložené ve výst.souborech SIT\_\*\*\*.ANA)
- ♦ průběžnou kontrolu překročení zadaných mezí (se zápisem do KONT.HLA a signalizací piktogramy)

<sup>1</sup> Definice elektrizační soustavy a jejich jednotlivých složek je podána v Popisu modelování [1]

<sup>2</sup> Stejně pojmenování má historický původ, neboť na počátku stál program pro dynamickou simulaci a později ostatní

<sup>3</sup> Popis jednotlivých vstupních souborů je proveden v Uživatelské příručce [2]

<sup>4</sup> načtení vstupních dat, výpočet počátečních podmínek a zadaných hodnot regulátorů

<sup>5</sup> stejný zásah provedený na množině větví nebo bloků

Kromě toho program MODES připravuje **vstupní soubory** pro přídatné programy, které vykonávají dodatečné funkce (tzv. „post-processing“) po skončení dynamické simulace. Můžeme je rozdělit do dvou skupin. První skupinu tvoří nástroje pro tvorbu grafů:

- \* ZOBRAZ - kreslení statických charakteristik asynchronních motorů a primární regulace
- \* GRAF - kreslení časových průběhů proměnných uložených v tzv. uživatelských **výstupních souborech**.

Druhou skupinu tvoří analytické nástroje pro vyhodnocení časových průběhů vybraných proměnných:

- ◇ ODCHYLKY- výpočet středních kvadratických odchylek a hodnot sekundární regulace P a f
- ◇ HARM\_ANA - vyhodnocení veličin (amplitudy a fáze) vynucených harmonických kvyů.

V této příručce se budeme zabývat popisem uvedených pomocných a přídatných programů.

## TISKOVÁ KONVENCE

V dokumentu jsou používány tyto písma a symboly:

**Tučný tisk** pro významné termíny definované v následující kapitole

**VELKA** písma pro jména **adresářů**, programů a označení kláves

**JMENO.PR** velké neproporcionální písmo pro jména **souborů**

**ARIAL** pro výpisy **souborů**

☞ tento symbol obsahuje upozornění na možné problémy při použití daného rysu programu

V programu je používána následující příponová konvence pro **vstupní/výstupní soubory**:

**DAT** pro **vstupní soubory** programu **MODES**

**VST** pro **vstupní soubory** pomocných a přídatných programů

**SAB** pro šablony **vstupních souborů** tvořené např. programem **PRIRAD**

**ORG** pro zálohu **vstupních souborů** tvořenou programy **UST** a **SCENARE**

**EKV** pro soubory tvořené programem **EKVIVAL**

**DTB** pro vstupní "databázové" soubory

**GRF** pro **výstupní soubory** **MODESu** určené jako vstup pro grafické zpracování přídatnými programy

**SRI** pro **výstupní soubory** **MODESu** určené jako vstup pro přídatné programy

**SRO** pro **výstupní soubory** přídatných programů

## DEFINICE POUŽITÝCH TERMÍNŮ

**Adresář** - část pevného disku vyhrazená pro programy a datové soubory. V hlavním adresáři jsou nainstalovány jednotlivé programy, v podadresářích **VST/VYST** jsou vstupní/výstupní soubory.

**Databáze** - forma **vstupního souboru** (má standardně příponu **DTB**). Obsahuje hlavičku s popisem a libovolný počet řádků se záznamy. Každý záznam je členěn do jednotlivých položek.

**Formulář** - forma **vstupního souboru** (má standardně příponu **DAT**). Je členěn do sekcí, každá sekce má hlavičku s popisem zadávaných údajů a nápovědnými informacemi.

**Katalog** - forma **vstupního souboru** (má standardně příponu **CAT**). Má obdobnou strukturu jako **databáze** s tím rozdílem, že je členěn do sekcí a počet záznamů je omezen

**Objekt** - prvek **ES**, který je modelován jako pojmenovaná entita. Jedná se o uzly, větve, bloky, oblasti, automatiky a stabilizátory.

**Scénář** - časový sled **zásahů**, uložený ve **vstupním souboru** **SCENAR.DAT**.

**Tabulka** - forma **vstupního souboru**. Má obdobnou strukturu jako **databáze** s tím rozdílem, že má pevně stanovený počet řádků. Každý řádek - záznam odpovídá jednomu prvku **ES**.

**Vstupní soubor** - obsahuje vstupní data programu (je uložen v podadresáři **VST**). Má formu **formuláře**, **tabulky**, **katalogu** nebo **databáze**.

**Výstupní soubory** - vznikají v podadresáři **VYST**. Jsou to hlášení programu a soubory s analýzou. V souborech s příponou **GRF** jsou uloženy statické charakteristiky primární regulace a as.motorů pro další zpracování do grafů. Dále se jedná o soubory s příponou **SRI**, určené pro zpracování v **přídatných programech**. Lze zadat 18 typů uživatelských **výstupních souborů** pro ukládání časových průběhů **proměnných**. Tyto **soubory** mají formu pro další zpracování v tabulkových procesorech.

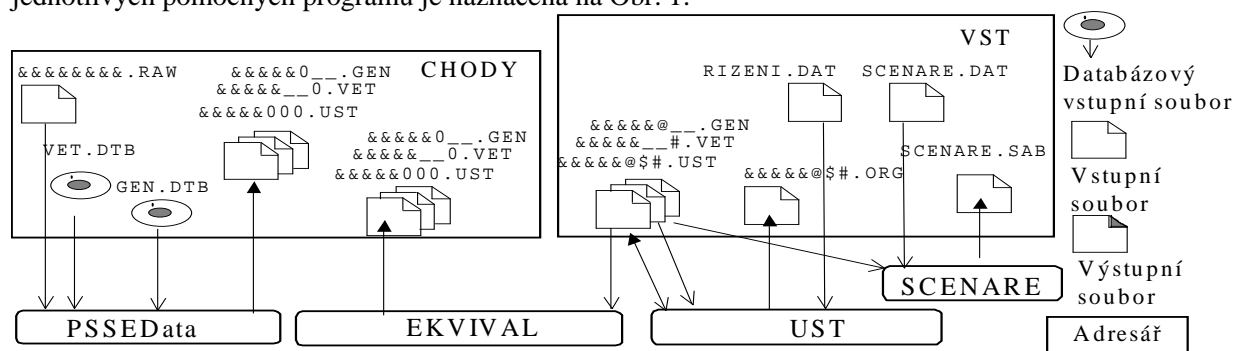
**Ustálený stav (US)** - stav soustavy, kdy všechny **proměnné** mají konstantní průběh. Jeho řešení se provádí programem na výpočet **ustáleného stavu** a je východiskem pro modelování **přechodných dějů**.

**Uživatelské rozhraní** - nadstavba programového balíku **MODES** pracující v prostředí **WINDOWS 3.1** a využívající služeb tohoto prostředí. Jedná se zejména o systém nápovědy, o přístup ke **vstupním a výstupním souborům** a spouštění jednotlivých pomocí ikon nebo menu.

**Zásah** - modeluje v programu záměrné **změny**. Mohou být zadávány před spuštěním výpočtu (formou **scénáře**), po spuštění výpočtu (aktivaci horkých kláves) nebo jsou aktivovány činnostmi **automatik**.

## 1. SOUVISLOST MEZI POMOCNÝMI PROGRAMY

Účelem pomocných programů je připravit vstupní datové soubory pro program MODES. Hierarchie jednotlivých pomocných programů je naznačena na Obr. 1:



**Obr. 1. Vzájemné vztahy mezi pomocnými programy balíku MODES**

V obrázku je naznačeno pomocí šipek, které **soubory** jednotlivé programy používají jako vstupní/výstupní. Vstupní soubory (s příponami UST, VET a GEN pro tzv. chod sítě a DAT pro soubory obsahující parametry výpočtu), které se nachází v podadresáři VST lze přímo použít programem MODES. Chody sítě, vytvořené programem PSEEData (ze zadaného souboru RAW) nebo programem EKVIVAL (z aktuální varianty chodu sítě) je možno použít pro tvorbu nového projektu nebo případu. K pohodlnému vytváření nových projektů/případů slouží uživatelské rozhraní MODMAN (z menu Projekt/Případ Nový).

Na vrcholku hierarchie stojí pomocný program PSEEData, který transformuje vstupní data ve formátu PSS/E<sup>1</sup> do **vstupních souborů** MODESu. Transformace má jednorázový charakter, provede se jednou a výsledné soubory, uložené v podadresáři CHODY mohou být použity k tvorbě nového případu z uživatelského rozhraní MODMAN.

O stupeň níže stojí pomocný program EKVIVAL, který slouží k zjednodušení modelované soustavy. Toto zjednodušení spočívá v tom, že ve vybraných oblastech se eliminují všechny uzly, kromě těch do kterých jsou připojeny bloky (zadané buď počtem fyzických bloků nebo s výkonem větším, než je zadaný vztažný výkon Sv). Detailní popis je uveden v jedné z následujících kapitol. Redukce má opět jednorázový charakter a výsledky jsou uloženy opět v adresáři CHODY, odkud mohou být použity pro tvorbu nového případu.

Pomocný program UST provede výpočet nového ustáleného stavu (např. po změně ve vstupních souborech chodu sítě). Jeho použití má opakovaný charakter. Program vždy vyzálohuje původní vstupní soubor UST.DAT (nebo variantu) do stejnojmenného souboru s příponou ORG a přepíše ho nově vypočítaným. Po dvojím výpočtu se tedy původní vstupní soubor UST.DAT přepíše.

Jestliže jsme při výpočtu ustáleného stavu podstatně měnili dodávaný výkon do uzlů, je nutné aktualizovat i vstupní soubor GEN.DAT, jinak program MODES může při inicializaci<sup>2</sup> hlásit porušení zadaných mezí.

Tvorbu hromadných scénářů pro zásah výpadku bloku, větve nebo zkratu provádí program SCENARE.

Všechny programy jsou v pracovním adresáři a jejich **vstupní/výstupní soubory** jsou v podadresářích VST/VYST. Vyjimky tvoří program TRANSF, který je umístěn ve zvláštním adresáři včetně svých podadresářů VST/VYST.

<sup>1</sup> mezinárodně uznávaný formát pro elektronickou výměnu dat pro výpočty ustáleného stavu („Raw data format“)

<sup>2</sup> postup inicializace je popsán v Příručce uživatele [2]

## 2. POMOCNÉ PROGRAMY

### 2.1 Transformace souborů RAW programem PSSEData

Pomocný program PSSEData umožňuje převod datového souboru ve formátu PSS/E (jméno maximálně 8 alfanumerických znaků a přípona RAW) do **vstupních souborů** chodu sítě programu MODES. Vstupní soubory typu RAW musí být v podadresáři CHODY, kam musí být vloženy (zkopírovány) uživatelem.

Program funguje pro verze programu PSS/E od 23 do 27. Podmínkou správné funkce je dodržení přesné syntaxe vstupního souboru RAW podle dokumentace programu PSS/E (Program Operational Manual –Volume I, Activity Read).

Jelikož formát PSS/E nezná jména větví, je možné je definovat v „databázi“ VET.DTB. Tento textový soubor uložený v podadresáři CHODY obsahuje jednak komentářové (slouží pouze uživateli) řádky začínající znakem apostrofu (‘) a jednotlivé záznamy ve tvaru:

Jm\_vetve Jm\_uzlupoc ‘Skutečné jméno ‘ Jm\_uzlukon ‘Skutečné jméno‘

Jm\_vetve ..... symbolické jméno větve jako objektu v programu MODES (8 znaků)

Jm\_uzlupoc, Jm\_uzlukon..... symbolická jména počátečního a koncového uzlu větve (8 znaků)

V apostrofech jsou skutečná jména uzlů (složí pro informaci uživatele).

Symbolická jména uzlů musí odpovídat jménům použitým ve vstupním souboru &&&&&&&.RAW. Pokud větev není v databázi nalezena, je jí přiřazeno standardní jméno VET\_000 a může být zeditována později uživatelem.

Jelikož formát PSS/E nezná jména bloků, je možné je definovat v „databázi“ GEN.DTB. Tento soubor je uložen v podadresáři CHODY. Soubor obsahuje informace o lokalizaci (přiřazení jmen **bloků** jménům **uzlů**), dostupnosti (možném počtu **bloků**) a další údaje. Struktura části tohoto souboru, týkající se generátorů je patrná z následujícího výpisu:

V souboru GEN.DTB jsou uloženy údaje o jednotlivých blocích

Ngen: počet bloku, Nmot: počet motoru:

```

136 0
*****
Pocet Mod  nazev  nazev   Sn    Ntmin Ntmax Xd"  Pt    Xt  Reg Min1 Max1 Min2 Max2  PRR
bloku bloku bloku  uzlu   [MVA] [MW]  [MW]  [p.j.]  [p.j.] [MW] [MW] [MW] [MW] [%Ntmax]
                                           Ngenx
2    1 'EDET' 'C3DET' 235.0 140  200  0.154 0.92 0.12 2 140 180 180 200 5

```

První položka (Počet bloků) udává disponibilní počet fyzických bloků<sup>1</sup>, které se mohou použít na pokrytí výkonu dodávaného do **uzlu**. Druhá položka (Mod bloku) určuje způsob určení výkonu **bloku**. Při volbě 0 se výkon **bloku** určí jako suma zdánlivých výkonů Sn nasazených fyzických bloků. Při volbě 1 se použije druhý způsob určení výkon počtem nasazených fyzických bloků. Další dvě položky definují řetězce jmen **bloku** a **uzlu**, kam je **blok** připojen. Sn určuje zdánlivý výkon jednoho fyzického bloku. Obdobně Ntmin-Ntmax určují regulační rozsah turbíny. Další tři položky určují v poměrných hodnotách rázovou reaktanci, převod a reaktanci blokového trafo (nula pokud je blokové trafo součástí modelu sítě). Zbývající údaje se týkají primární a sekundární regulace. Reg určuje počet regulačních pásem pro sekundární regulaci (lze zadat maximálně dvě regulační pásma). Hodnoty Mini-Maxi definují minimální a maximální výkon v daném pásmu (i=1,2). Poslední údaj definuje velikost požadované primární regulační rezervy v procentech maximálního regulačního výkonu turbíny Ntmax.

Za záznam bloku je možné přidat klíčové slovo Alias bezprostředně (bez mezery) následované alternativním jménem uzlu v apostrofech.

Databáze GEN.DTB obsahuje i úsek motorů, jehož struktura je patrná z následujícího výpisu:

```

pocet. Mod  jmeno  jmeno  Sn    Kzat  Rv  Xv  Cos*Eta  Part  Kzmin Kzmax
motoru  motoru  uzlu   [MVA] [p.j.] [p.j.] [p.j.] [p.j.] {0-120} [p.j.] [p.j.]
                                           Nmotx
1    1 'CER_A' 'VL_SPA' 6.000 0.9  0.0  0.0  0.85  100  0.6  1

```

První dvě položky v záznamu by měly být 1. Parametry Kzmin-Kzmax určují povolený rozsah zatěživatele (poměr skutečného a jmenovitého zatížení motoru). Pokud je dodávaný výkon v **uzlu** záporný a

<sup>1</sup> Počtem fyzickým bloků se rozumí počet soustrojí, které se pak v programu modelují jako ekvivalentní blok

v **databázi** jsou k danému **uzlu** nalezeny motory, program postupně přiřazuje motory z **databáze**, až je výkon pokryt. Zároveň se vypočítají zatěživatelé, tak aby platila výkonová rovnováha (příkon motorů se rovnal záporně vzaté dodávce do **uzlu**).

Program vyřazuje:

- tzv. ekvivalentové větve (s parametrem CKT >90), v kterých tok nepřevyšuje 10 MW a 10 MVar
- izolované uzly (s parametrem KOD=4) nebo do vyřazených ekvivalentních větví
- záznamy s chybou formátu dat.

O všech vyřazených uzlech je hlášení ve výstupním souboru PSSEData.hla v podadresáři CHODY (dostupný také příkazem **Hlášení/Transformace dat z PSSE**). Ve verzi MODES 2.2/7, která neobsahuje model stejnosměrných vedení (nebo spojek) musí být činný a jalový výkon odebíraný stejnosměrným vedením doplněn do **vstupního souboru** UST.DAT ručně.

U napájecích uzlů (specifikovaných v úseku generátorů) se kontroluje rozsah jalového výkonu. V případě, že neplatí nerovnost  $Q_{min} < Q_G$  nebo  $Q_G < Q_{max}$ , program vypočítá meze ze vztahů  $Q_{min} = \min[\min(1.05 \cdot Q_G, 0.95 \cdot Q_G), -0.48 \cdot P_G]$  nebo  $Q_{max} = \max[1.05 \cdot Q_G, 0.62 \cdot P_G]$  neboli určí rozsah jaloviny buď z definované dodávky jalového výkonu  $Q_G$  nebo z činné dodávky  $P_G$  násobené tangentou odpovídající účinnosti 0.85 pro  $Q_{max}$  a 0.9 pro  $Q_{min}$ . U napájecích uzlů, pro které je ve vstupních datech  $Q_T = Q_B$  (obě meze jaloviny jsou stejné) převádí se do výstupních souborů  $Q_{min}$  a  $Q_{max}$  nulové, aby nedocházelo při následném výpočtu chodu sítě k degeneraci napájecích uzlů na odběrové, což vede ke zpomalování výpočtu, protože se musí provádět triangularizace Jacobiho matice.

Program přiřadí **uzlům**, které najde v **databázi**, příslušný počet fyzických bloků z **databáze** a zatíží je tak, aby pracovní bod ležel v dovoleném rozsahu. Dovolенý rozsah tvoří jedno z dvou možných sekundárních regulačních pásem případně zmenšené shora/zdola o primární regulační rezervu (je-li předepsána). Pokud je Reg=0 (bloky nejsou v sekundární regulaci) tvoří dovolенý rozsah přímo regulační rozsah  $N_{tmin} - N_{tmax}$  opět zmenšený o primární regulační rezervu. Pokud k **uzlu** chybí v **databázi** záznam a jedná se o tzv. napájecí uzel (s nenulovým  $Q_{min}/Q_{max}$ ) o dodávaném výkonu větším než 200 MW, použijí se standardní **bloky** 235 MVA s jménem NONAME a. regulačním rozsahem 100-200 MW. V případě, že úloha nemá řešení (dodávka v **uzlu** je nízké nebo naopak vysoké pro nasazení bloku v regulačním rozsahu) program vypíše hlášení do **výstupního souboru** PSSEData.hla a blok v tabulce bloků vypne (Stav=0). Do jednoho **uzlu** lze připojit až dvě elektrárny, přičemž se bloky první (uvedené v GEN.DTB dříve) zatěžují přednostně.

Během transformace program generuje vstupní soubor s generátory, přičemž do nich zařazuje všechny uzly, které našel v databázi se do generátorických uzlů zařazují uzly s výkonem větším než 200 MW (vyřazení malých výkonů).

Přídavný program PSEEData je nainstalován v pracovním adresáři MODESu a spouští se z uživatelského rozhraní MODMAN z menu příkazem **Nástroje/Konverze dat z PSSE**. Po spuštění je nutno zadat jméno vstupního souboru typu RAW (jako pomůcku program vypisuje všechny soubory tohoto typu v podadresáři CHODY). Pak je nutno zadat pětímístné jméno výstupního chodu sítě &&&&&.

Základní variace chodu sítě (&&&&&\_0.VET, &&&&&0\_.GEN a &&&&&000.UST) jsou uloženy v podadresáři CHODY. Výstupní soubory chodu sítě se dají použít pro tvorbu nového případu příkazem **Případ/Nový** kde se dialogem vybere příslušný chod sítě ze skladiště chodů.

Transformační program rovněž vytváří výchozí databázi modelů bloků (pojmenovanou stejně jako jméno chodu s příponou DTB - &&&&&.DTB) a výchozí katalog typových parametrů (pojmenovanou stejně jako jméno chodu s příponou CAT - &&&&&.CAT). Podmínkou je, aby příslušný záznam se jménem bloku a uzlu byl v databázi BLOK.DTB uložen v podadresáři DATABASE. Při sestavování nového projektu (pomocí editoru projektů dostupného z menu příkazem **Projekty/Nový**) je pak možno vybrat jak vytvořený chod, tak připravenou databázi a katalog.

## 2.2 Přiřazení bloků do uzlů

Součástí balíku MODES může být pomocný program PRIRAD, který umožňuje zmechanizovat proces p5i5azen9 blok; do uzl;.. Tuto činnost provádějí i transformační programy.

Znalosti o lokalizaci (přiřazení jmen **bloků** jménům **uzlů**), disponibilitě (možném počtu **bloků**) a další údaje musí být uloženy v **databázi** GEN.DTB. Program PRIRAD pak přiřadí napájecím **uzlům** (to je těm, které mají nenulové meze jalové dodávky  $Q_{min}$  nebo  $Q_{max}$ ) příslušný počet **bloků** z **databáze** a zatíží je tak, aby pracovní bod ležel v dovoleném rozsahu. Dovolený rozsah tvoří jedno z dvou možných sekundárních regulačních pásem případně zmenšené shora/zdola (pokud je předepsána primární regulační rezerva). Pokud k příslušnému **uzlu** chybí v **databázi** záznam, použijí se standardní **bloky** 200 MW. Program zjistí potřebný počet bloků, nutných pro pokrytí dodávaného výkonu, přičemž bere v úvahu až dvě regulační pásma, které mohou být zúženy o primární regulační rezervu. V případě, že úloha nemá řešení (zatížení v **uzlu** je nízké nebo naopak vysoké) program vypíše hlášení do **výstupního souboru** PRIRAD.HLA. Do jednoho **uzlu** lze připojit až dvě elektrárny, přičemž se bloky první elektrárny (uvedené v databázi GEN.DTB dříve) zatěžují přednostně. Toto pravidlo lze přepsat zadáním participačních koeficientů.

V případě, že program PRIRAD pracuje samostatně vytvoří šablonu **vstupního souboru** GEN.SAB, kterou uloží do podadresáře VST. Pak si uživatele přejmenuje soubor na GEN.DAT, který lze pak použít jako součást výchozího **chodu**. V případě transformačních programů je soubor s **tabulkou bloků** uložen přímo v podadresáři CHODY.

Pokud je dodávaný výkon v **uzlu** záporný a v **databázi** jsou **uzlu** připojeny motory, program postupně přiřazuje motory z **databáze**, až je výkon pokryt. V **databázi** lze předepsat participační koeficient, předepisující v jakém poměru příslušný motor přebírá případný deficit výkonu (rozdíl mezi sumárním příkonem motorů připojených do **uzlu**<sup>1</sup> a dodávkou do uzlu<sup>2</sup>). Toto je první způsob zadávání asynchronní zátěže - tzv. blokový. Používá se v případě, že známe parametry jednotlivých motorů, např. v případě modelování vlastní spotřeby bloku.

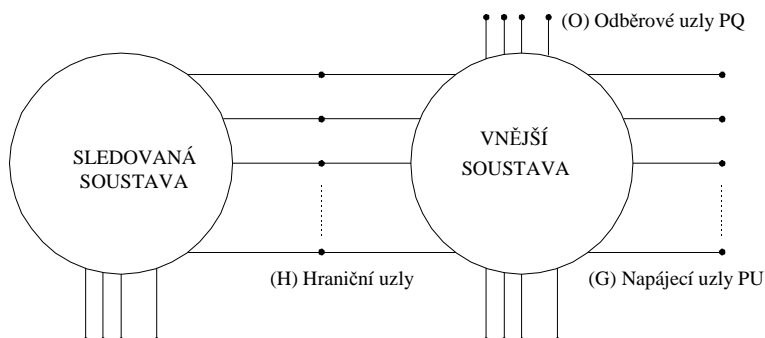
---

<sup>1</sup> součinů jmenovitého výkonu, účinníku, účinnosti a zatěžovatele

<sup>2</sup> Pdod s obráceným znaménkem z tabulky UST.DAT

## 2.3 Redukce počtu uzlů programem EKVIVAL

Pomocný program EKVIVAL umožňuje zmenšení modelované soustavy tím, že uzly vybraných oblastí jsou vyřazeny Gaussovou eliminací. Vybrané oblasti jsou definované v tabulce uzlů, v úseku oblastí, kde se zadá parametr 0 pro oblasti, jejichž uzly mají být eliminovány a 1 pro oblasti, které mají být zachovány beze změny. Hraniční uzly eliminovaných oblastí přilehlých k oblastem zachovaným eliminovány nebudou. V následujícím obrázku je sledovaná soustava vymezena oblastmi s parametrem 1 a vnější soustavou oblastmi s parametrem 0.



Obr. 2. Rozdělení soustavy na sledovanou a vnější část

Napájecí uzly se definují tak, že jsou do nich připojeny bloky:

- zadané v tabulce bloků počtem (celým číslem)
- zadané v tabulce bloků výkonem  $S_n$  (s desetinou tečkou) větším, než je mezní výkon.

Mezní výkon je roven vztažnému výkonu  $S_v$ , který se zadává v hlavičce tabulky uzlů.

Přídatný program EKVIVAL je nainstalován v pracovním adresáři MODESu a spouští se z uživatelského rozhraní MODMAN z menu příkazem **Nástroje/Ekvivalentování**. Po spuštění je nutno zadat pětímístné jméno výstupního chodu sítě &&&&&. (jako pomůcku program vypisuje všechny soubory tohoto typu v podadresáři CHODY).

Základní variace chodu sítě (&&&&&\_0.VET, &&&&&0\_.GEN a &&&&&000.UST) jsou uloženy v podadresáři CHODY. Výstupní soubory chodu sítě se dají použít pro tvorbu nového případu příkazem **Případ/Nový** kde se dialogem vybere příslušný chod sítě ze skladiště chodů. Do podadresáře VYST program uloží soubor s hlášením EKVIVAL.HLA, který je dostupný příkazem **Hlášení/Transformace z PSSE**.



## 2.4 Výpočet ustáleného stavu (US) pomocí programu UST

### 2.4.1 Spuštění programu UST

Spuštění se provede příkazem UST, který zajistí zálohování původního **Est.souboru** UST.DAT (záloha má příponu ORG) a provede vlastní výpočet US, jehož výsledek se uloží v podadresáři VST do souboru UST.DAT, který lze bez úprav použít programem MODES. **Vstupní soubory** tvoří zmíněný soubor UST.ORG, soubor VET.DAT a soubor RIZENI.DAT s parametry výpočtu. Hlášení o chybách a diagnostice jsou ve **výstupních souborech** CHOD.HLA a UST\_DIAG.HLA, které jsou uloženy v podadresáři VYST.

### 2.4.2 Ladění ustáleného stavu

Podstatou výpočtu US je zjištění napětových poměrů. Při tom musí být dodrženy následující základní podmínky:

- a) celková výkonová vyrovnanost - znamená, že součet výkonů dodávaných do **uzlů** se rovná odebíraným výkonům a ztrátám ve **větvích** - tím je zaručena i jmenovitá frekvence. **Uzel**, který hradí výkonový deficit se nazývá bilanční. V programu UST je tento **uzel** zároveň referenčním - fázor jeho napětí je umístěn do reálné osy souřadného systému **sítě** a fáze napětí ostatních **uzlů** se odčítají od této osy.
- b) dodržení zadaného činného dodávaného výkonu, amplitudy napětí a rozsahu jalových výkonů v napájecích **uzlech**
- c) dodržení zadaných činných a jalových dodávek v odběrových **uzlech**
- d) dodržení zadaných činných a jalových odběrů ve všech **uzlech**.

Postup při ladění může být následující:

- A) pomocí textového editoru upravíme **vstupní soubory** UST.DAT a VET.DAT, přičemž musí být dodržena struktura souboru t.j.: počet řádků hlavičky, skutečný počet řádků s údaji (musí odpovídat zadanému), počet a formát údajů na řádku (nelze žádný údaj vynechat a je třeba dodržet pořadí). Vzorové **vstupní soubory** jsou součástí dodávky. **Soubor** UST.DAT obsahuje:

- názvy **uzlů**
- přiřazené čísla **oblastí**
- jmenovitou hodnotu napětí (napětíovou hladinu)
- zadanou hodnotou napětí (to je důležité pro bilanční a napájecí **uzly**, pro ostatní lze zadat libovolné číslo, neboť správná hodnota je výsledkem výpočtu - totéž platí pro fázi napětí ve všech **uzlech**)
- výkony **zátěží** (program UST předpokládá, že výkony jsou konstantní - nezávislé na napětí **uzlu**)
- výkony dodávané do **uzlu** (činný výkon v bilančním uzlu bude výsledkem výpočtu, u ostatních uzlů budou dodávané činné výkony drženy na zadané hodnotě; jalové výkony bilančního a napájecích **uzlů** budou výsledkem výpočtu; u odběrových **uzlů** budou udržovány na zadané hodnotě)
- kompenzační výkon (tento výkon má jalový charakter, přičemž kladná hodnota značí připojenou kapacitu a záporná připojenou tlumivku; jelikož velikost kompenzačního výkonu se tentokrát mění s kvadrátem napětí, platí zadaná hodnota pro jmenovitou hodnotu příslušné hladiny)
- rozsah jalového výkonu pro bilanční a napájecí **uzly** (tyto hodnoty hrají ve výpočtu US důležitou roli, jestliže program UST vypočítá hodnotu jalového výkonu mimo tyto meze, pak v případě napájecího **uzlu** je tento zařazen mezi odběrové s příslušným zadaným mezním jalovým výkonem a v případě bilančního uzlu výpočet nezkonverguje a hlásí překročení maximálního počtu iterací).

**Soubor** VET.DAT obsahuje:

- jména **větví** a čísla počátečního a koncového **uzlu**
- parametry **větví** (význam je v [1])
- velikost bočniců připojených do počátečního a koncového **uzlu**
- mezní proud (zdánlivý výkon pokud se jedná o trafo).

Při zadávání údajů o trafech je vhodné, aby přepínání odboček bylo umístěno na straně počátečního **uzlu**. Pak totiž není nutné uvažovat změnu parametrů trafa při změně odboček. Pouze je nutné, aby počáteční **uzel** předcházel koncový ve **vstupním souboru** UST.DAT. Parametry traf jsou přepočítány na napětí koncového uzlu (viz [1]).

- B) upravíme **soubor** RIZENI.DAT s parametry výpočtu v úseku určeném pro program UST:

**Soubor** RIZENI.DAT obsahuje:

- dovolenou bilanční nerovnováhu výkonů v uzlu (v MVA)
- maximální dovolený počet iterací
- zadání výpočtu jakobiánu v každé iteraci (Jacob=1)
- způsob inicializace hodnot napětí v první iteraci - buď tzv. "flat start" se zadanými moduly napětí v napájecích **uzlech**, jmenovitými hodnotami napětí v odběrových **uzlech** a nulovými fázemi napětí ve všech **uzlech** (Ifat=1) nebo použití hodnot z předchozího výpočtu.
- požadavek na diagnostiku iteračního procesu (Diagn=1), kdy se do **výstupního souboru** UST\_DIAG.HLA vypisuje hlášení o výkonové nerovnováze v jednotlivých **uzlech** během iterací



Při prvním výpočtu se zadává "flat start" - tedy Ifat=1. Při volbě výpočtu jakobiánu v každé iteraci program potřebuje menší počet iterací na dosažení požadované přesnosti. Samotný výpočet jakobiánu je časově náročný, takže celková doba výpočtu může být delší než bez přepočítávání jakobiánu. Proto tento způsob volíme v případě, že výpočet s volbou Jacob=0 nezkonverguje. Obvykle není potřeba provádět diagnostiku (Diagn=0).



Pokud výpočet nekonverguje, zadá se diagnostika pomocí parametru Diagn=1. Ve **výstupním souboru** UST\_DIAG.HLA se pak nalezne příčina. Jsou to **uzly** jejichž výkonová nerovnováha nekonverguje. Může se jednat například o **uzly** propojené větví o malé impedanci (např. spínač přípojnic). Pak je nutné spínač odstranit a přípojnice modelovat jako jeden **uzel**.

- C) provedeme první výpočet a zkontrolujeme, jestli vypočítaná činná dodávka v bilančním **uzlu** nepřesahuje dovolené meze (tyto meze nejsou v programu zadávány a proto je nutné, aby kontrolu prováděl uživatel). Pokud jsou tyto meze překročeny, rozdíl mezi dovolenou a vypočítanou dodávkou je nutno přerozdělit mezi **napájecí uzly** a provést nový výpočet. Obdobná kontrola je nutná pro jalovou dodávku v bilančním uzlu. V tomto případě jsou meze jalové dodávky zadávány a jestliže jsou překročeny, výpočet nezkonverguje. Na začátku je vhodné zadat širší meze a postupně je zužovat, až odpovídají reálným.



Změny vstupních dat týkající se **uzlů** se zadávají do **souboru** UST.DAT. Program UST zajistí zálohování **souboru** UST.DAT do **souboru** UST.ORG a výpočet nového **US**. V každém případě je výsledek výpočtu **US** uložen v **souboru** UST.DAT.

- D) pokud v bilančním **uzlu** vychází jalový odběr mimo reálné meze, můžeme požadavek na jalovinu změnit zadaným napětím - snížením při velké jalové dodávce (kladný výkon) a zvýšením při záporném výkonu. Napětí by ovšem mělo být v mezích  $\pm 5\%$  Un. Druhou možností je přidání kompenzačního výkonu do **uzlu**. Bude se jednat o zadání kladné hodnoty pro nízké hodnoty napětí (resp. pro velké kladné hodnoty jalové dodávky v bilančním **uzlu**) a záporné hodnoty pro opačné případy. Třetí možností je změna převodů regulačních traf, čímž se dosáhne přerozdělení jalového výkonu mezi jednotlivé napěťové hladiny.
- E) podobný proces se provede pro napájecí **uzly**, až se dostane jalová dodávka do zadaných mezí. V případě, že požadovaný jalový výkon vybočí z dovolených mezí, napájecí **uzel** se změní na odběrový se zadanou jalovou dodávkou rovnou překročené mezi. Zároveň je zapsáno hlášení do **výstupního souboru** CHOD.HLA.
- F) s odběrem a dodávkou odběrových **uzlů** se běžně během ladění nemanipuluje. V krajním případě lze tyto veličiny měnit, pokud se nepodaří předchozími kroky dosáhnout pokrytí zátěže.

Popsaný postup je jen velmi zjednodušeným odrazem procesu řízení rovnováhy mezi výrobou a spotřebou elektrické energie, tak jak probíhá v praxi. Tento proces probíhá zhruba ve čtyřech úrovních: primární, sekundární a terciární regulaci a plánování provozu. Jako pátou úroveň (správněji možná nultou úroveň) tvoří havarijní odlehčování zátěže.

Primární regulace spočívá v tom, že zdroje udržují zadanou hodnotu svorkového napětí a činného výkonu. V praxi se používá tzv. statická regulace, kdy zadaná hodnota není konstantní, ale mění se v závislosti na jalovém zatížení a odchylkách frekvence v síti.

Sekundární regulace P/f spočívá v tom, že centrální regulátor frekvence a předávných výkonů mění zadanou hodnotu výkonu regulačních bloků. Zde již nacházíme analogii mezi touto regulací a bodem C) ladění **US**, kdy deficit činného zatížení se rozděluje na bilanční a napájecí **uzly**. Podle jakých pravidel se toto rozdělení provede nebylo specifikováno. V praxi toto rozdělování provádí právě centrální regulátor. Nejjednodušší pravidlo je proporcionální rozdělení nebo rozdělení pomocí participačních koeficientů.

Sekundární regulace U/Q zajišťuje udržování napětí v pilotních **uzlech**. Zde je patrná souvislost pilotních **uzlů** s napájecími **uzly** při výpočtu **US**. Zhruba řečeno tato regulace předepisuje hodnoty jalového výkonu regulačních bloků a mění převody regulačních traf. Případně může připínat kompenzační prostředky (kondenzátorové baterie nebo tlumivky). Je zde patrná souvislost mezi body D) a E) ladění **US**.

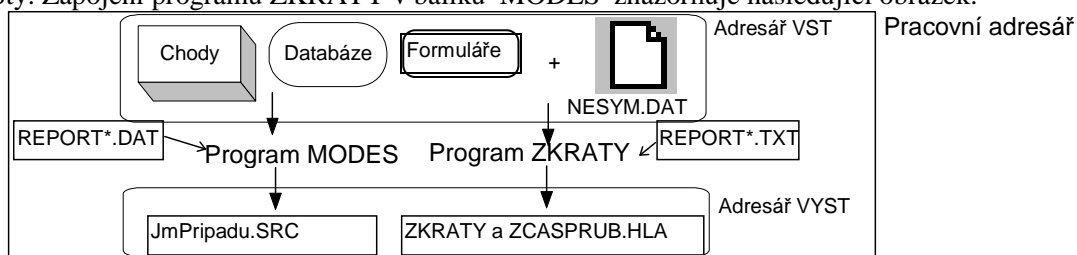
Terciární regulace na základě optimalizace (ekonomických kritérií) předepisuje zadané hodnoty napětí v pilotních **uzlech** a přerozděluje činnou dodávku regulačních bloků. Výsledkem je optimální **US**.

Krajním případem vyrovnání výkonové bilance je odlehčování zátěže, odpovídající bodu F).

## 2.5 Výpočet časového průběhu a char.hodnot zkratového proudu ZKRATY

### 2.5.1 Účel programu

Program slouží k výpočtu veličin zkratového proudu. Na rozdíl od běžných programů počítá nejen počáteční rázový zkratový proud  $I''_k$ , ale i časový průběh souměrné složky zkratového proudu o základním kmitočtu. Velikost periodické složky o dvojnásobném kmitočtu se zanedbává ( $X_d'' = X_q''$ ). Stejnoseměrná složka příspěvků zkratového proudu se aproximuje analytickým řešením okamžitých proudů rozšířeného aktivního dvojbranu za předpokladu vzniku její maximální možné hodnoty. Zapojení programu ZKRATY v balíku MODES znázorňuje následující obrázek:



- Program se nachází v pracovním adresáři - to je tam, kde je balík programů MODES nainstalován
- Program využívá stejné soubory chodů a databází jako MODES
- Z formulářů využívá program ZKRATY soubory SCENAR a RIZENI.
- Data pro nesymetrické poruchy jsou uloženy v souboru NESYM.DAT.
- Výstupy programu jsou uloženy v souborech ZKRATY.HLA a ZCASPRUB.HLA (časový průběh 3f.zkrat.proudu)
- Program ZKRATY je aplikací pro WINDOWS (na rozdíl od dosovského programu MODES) a využívá pomocné soubory REPORT1-REPORT3.TXT (na rozdíl od MODESu využívajícího stejnojmenné soubory s příponami DAT).

### 2.5.2 Instalace programu

Program se nainstaluje prostým zkopírováním souboru ZKRATY.EXE z media do pracovního adresáře MODESu. Pokud je součástí servisního balíčku, rozbalí se balíček v pracovním adresáři.

### 2.5.3 Předpoklady fungování programu

1. Všechna vstupní data musí být připravena a odladěna programem MODES
2. Parametry pro nulovou složku musí být ve vstupním souboru NESYM.DAT, který obsahuje tabulky větví, generátorů motorů a zátěží. Soubor je společný i pro program MODES, takže jeho obsah je popsán v Uživatelské příručce programu MODES.

Při sestavování schéma sítě pro netočivou složku je potřeba postupovat obezřetně v tom smyslu, že je nutno zadávat jen ty větve, kterými se netočivá složka může uzavírat. Jelikož netočivá složka se uzavírá hlavně přes uzemněný střed hvězdy traf při druhém vinutí zapojeném do trojúhelníka, plyne z toho závěr: nezadávat větve připojené na stranu trojúhelníka, která je pro nulovou složku neprůchodná. Je-li taková větev zadána, program vzhledem k použité metodice (viz dále) není schopen spočítat inverzi síťové admitanční matice  $Z_k$  a tudíž náhradní impedanci  $Z_{k0}$  pro netočivou složku. Na tuto skutečnost program upozorní hlášením ve výstupním souboru AKCE.HLA (dostupný příkazem Hlášení/Akce) a zároveň do souboru ZKRATY.HLA (dostupný příkazem Hlášení/ Výpočet zkratových proudů) vypíše hodnotu  $z_k(0) \rightarrow \infty$ .

Při zadávání dat pro nulovou složku platí zásada, že pokud prvek nemá záznam v souboru NESYM.DAT tvoří pro nulovou složku nekonečnou impedanci a nulová složka se jím neuzavírá. Transformátory se zapojením typu 0, tedy bez uzemnění uzlu není třeba zadávat stejně jako generátory (případně syn.motory), které nemají účinně uzemněnou nulu.

3. ve vstupním souboru SCENARE se zadá jakýkoliv zkrat na vedení, jak pro program MODES. Vzdálenost zkratu se zvolí 0 nebo 100 podle toho, jestli počítáme zkrat v počátečním nebo koncovém uzlu. Vypnutí zkratu se zadá klíčovým slovem CLER. Uživatel tedy definuje postiženou větev, uzel se zkratem a dobu jeho trvání tk (rozdíl časů zadání zásahů CLER a FOUL). Program ZKRATY postupně spočítá všechny typy zkratů, které se v programu MODES zadávají následujícími klíčovými slovy:

FOUL -třífázový zkrat

FSLG - jednofázový zkrat („Single Line Ground“)

FDLG - dvoufázový zkrat („Double Line Ground“)

F\_LL - dvoufázový zkrat („Line to Line“)

U jedno, třífázového a u dvoufázového zemního zkratu lze zadat velikost reaktance  $X_G$  připojované mezi místo zkratu a zem. U dvoufázového zkratu lze zadat reaktanci  $X_F$  připojovanou mezi fázemi.



Hodnoty  $\underline{I}_R$  se zjistí z hodnot v sousledné složkové soustavě pomocí vztahů  $\underline{I}_R = \underline{I}_R + \underline{C}_i R * \underline{I}_{zkr_i}$  pro hodnotu ve fázi a (jednofázový zkrat) a  $\underline{I}_R = a^2 \underline{I}_R + \underline{C}_i R * \underline{I}_{zkr_i}$  pro hodnoty ve fázi b (dvoufázové zkraty).  $\underline{C}_i R$  je odvozeno v [5].

Typ zkratu	jednofázový ve fázi a	dvoufázový ve fázi b	dvoufázový zemní ve fázi b
$\underline{C}_i R$	$\underline{C}_2 + \underline{C}_0$	$-a * \underline{C}_2$	$[-\underline{C}_0 \underline{Z}_{K2} - a \underline{C}_2 (\underline{Z}_{K0} + 3 \underline{Z}_G)] / (\underline{Z}_{K2} + \underline{Z}_{K0} + 3 \underline{Z}_G)$

7. Programu určí sumární impedanci  $\underline{Z}_U$ . Pokud jsou do postiženého uzlu připojeny bloky, odečtenou se jejich admitance  $\underline{Y}_{ui} = 1 / \underline{Z}_{ui}$  od sumární admitance  $\underline{Y}_u = 1 / \underline{Z}_u$ . Zbytková impedance je pak  $\underline{Z}_{uu} = 1 / (1 / \underline{Z}_u - 1 / \underline{Z}_{u1} - \dots - 1 / \underline{Z}_{uin} - \dots - 1 / \underline{Z}_{un})$
8. Spočítají se hodnoty časové konstanty stejnosměrných složek  $T_a = X / 314 R$ , kde X a R jsou imaginární a reálná složka impedance cesty proudu podle následující tabulky:

Proud	IZKR	IR a IRQ	IR pro jednostranné odpojení zkratu z blízkého uzlu	IRQ pro jednostranné odpojení zkratu z vzdáleného uzlu	IQQ	IQi
Zcesty	$\underline{Z}_{K1} + \underline{Z}_d$	$\underline{Z}_S + \underline{Z}_d + \underline{Z}_E // \underline{Z}_L$	$\underline{Z}_L + \underline{Z}_d + (\underline{Z}_E + \underline{Z}_U) // \underline{Z}_S$	$\underline{Z}_d + \underline{Z}_E + \underline{Z}_S$	$\underline{Z}_{UU} + \underline{Z}_d$	$\underline{Z}_{Ui} + \underline{Z}_d$

Pokud není připojen žádný blok je  $\underline{Z}_{UU} = \underline{Z}_U$

Pokud vychází  $R=0$  (v soustavě nejsou zadány odpory větví ani generátorů) použijí se následující implicitní hodnoty podle [4]. Je-li do postiženého uzlu připojen blok, určí se  $T_a$  podle jmenovitém výkonu  $P_{gn}$

$P_{gn}$	$\geq 500$ MW	$\geq 110$ MW	$\geq 55$ MW	$< 55$ MW
$T_a$ [s]	0.4	0.3	0.18	0.161

Není-li do postiženého uzlu připojen blok, určí se  $T_a$  podle vztažného napětí postiženého uzlu  $U_v$

$U_v$	$< 0.4$ kV	$< 110$ kV	$\geq 110$ kV
$T_a$ [s]	0.01	0.02	0.03

9. Okamžitá hodnota zkratového proudu a jeho příspěvků se aproximuje vztahem:

$$i_K(t) = \sqrt{2} [i_{-K}(t) \cos(\omega(t-t_0)) - I_K'' e^{-(t-t_0)/T_a}] \quad I_K'' = i_{-K}(t=t_0) \quad K = ZKR, R, Q, RQ, QQ, Q1 \dots Qn$$

$t_0$  je okamžik vzniku zkratu,  $i_{-K}(t)$  je souměrná (střídavá) složka zkratového proudu spočítaná simulačním výpočtem a  $T_a$  příslušná časová konstanta určená podle bodu 8.

Pro trojfázové kovové zkraty ( $\underline{Z}_F=0$ ) se určí alternativní průběh příspěvků procházejících dvěma impedancemi vztahem:

$$i_K(t) = \sqrt{2} [i_{-K}(t) \cos(\omega(t-t_0)) - I_K'' e^{-\lambda(t-t_0)} f\{\alpha(t-t_0) + \psi_K\} / f\{\alpha(\psi_K)\}] \quad I_K'' = i_{-K}(t=t_0) \quad K = R, RQ \quad f\{x\} = \sinh\{x\} \text{ nebo } \cosh\{x\}$$

parametry  $\lambda$ ,  $\alpha$  a  $\psi$  se určí z hodnot ekvivalentu:

$$\lambda = [L_L R_E + L_E R_L + L_S (R_E + R_L) + R_S (L_E + L_L)] / (2 L_E L_L \sigma) \quad \alpha = \sqrt{\{[(R_L R_E + R_S (R_E + R_L)) / (L_E L_L \sigma)] - \lambda^2\}}$$

Oba aproximované průběhy  $i_K(t)$  předpokládají vznik maximální stejnosměrné složky.

10. Z analytického průběhu okamžité hodnoty zkratových proudů  $i_K(t)$  se spočítají nárazový maximální proud, ekvivalentní oteplovací proud, vypínací proud a jeho stejnosměrná složka podle vztahů:

$$I_{km} = \max\{i_K(t)\} \text{ pro } t = \langle t_0, t_0 + t_k \rangle \quad t_0 \text{ okamžik vzniku zkratu a } t_k \text{ je doba trvání zkratu}$$

$$I_{ke} = \sqrt{(Q/t_k)} \quad Q = \int_{t_0}^{t_0+t_k} i_K(t)^2 dt \text{ pro } t = \langle t_0, t_0 + t_k \rangle$$

$$I_{vyp} = i_{-K}(t_0 + t_k)$$

$$i_{avyp} = (100 * I_K'' e^{-(t_0+t_k)/T_a} / I_{vyp})$$

Podrobnosti o výpočtu veličin zkratového proudu jsou uvedeny v příloze.

### 2.5.5 Spouštění programu

- program spustíme z rozhraní MODMAN příkazem Nástroje/Zkraty nebo příslušným tlačítkem na liště.
- alternativně je možno spustit řádkovým příkazem ZKRATY.

### 2.5.6 Výsledky zkratového výpočtu

Výsledky výpočtů jsou uloženy ve výstupním souboru ZKRATY.HLA dostupném z rozhraní

MODMAN příkazem Hlášení/ Výpočet zkratových proudů. Soubor obsahuje tři části:

- hlavičku s popisem chodu sítě (první komentářový řádek v tabulce uzlů), názvem postiženého uzlu a hodnotami  $\underline{Z}_K$  náhradních zkratových impedancí pro jednotlivé složkové soustavy
- výpis počátečních hodnot v čase  $t_0$  pro jednotlivé zkraty, jehož struktura je v ná

Dodatečná impedance připínaná při nesymetrických poruchách mezi místo poruchy a zem v sousledné složkové

Vypis hodnot pri jednofazovom zkratu (S\_L\_G)  $Z_d = (3.101413, 31.21154)$

Data uzlu Príspevky vetvi

Složka	Uzel	U[-]	argU[deg]	I[kA]	argI[deg]	Vetev	Uzel	U[-]	argU[deg]	I[kA]	argI[deg]
Faze a	NODE_3	0.000	0.0	7.94	-90.0	a	----				
1-Pos.		0.546	0.0	2.65	-90.0	b	---				
2-Neg.		0.454	-180.0	2.65	-90.0	c	----				
0-Zer.		0.092	-180.0	2.65	-90.0	g	+				

Fázové hodnoty napětí v uzlu a sumární rázový

Složkové hodnoty napětí v uzlu a sumárního rázový

E" a vstříkovaný proud do postiženého uzlu

----	E" _GEN	1.002	0.0	1.65	-90.0	1-Pos.
3_1	NODE_1	0.208	0.0	0.93	-90.0	Faze a
		0.649	0.0	0.46	-90.0	1-Pos.

Grafické znázornění postižené  
Fázové napětí ve vzdáleném uzlu postižené

Fázový proud <I ze vzdáleného uzlu postižené

Složkové napětí a proudy ve vzdáleném			0.350	-180.0	0.46	-90.0	2-Neg.
			0.092	-180.0	0.00	0.0	0-Zer.
Seznam větví připojených do postiženého		2_3A	NODE_2	0.574	0.0	0.27	-90.0 1-Pos.
uzlu se jménem vzdáleného uzlu		2_3B	NODE_2	0.574	0.0	0.27	-90.0 1-Pos.

V levé části výpisu (označené jako data uzlu) jsou vypsány napětí a poruchový proud (t.j. suma proudů přitékajících do místa poruchy ze strany postiženého uzlu a vzdáleného uzlu) a to fázové i složkové hodnoty. Výpis je doplněn grafickým znázorněním postižené fáze. Pro trojfázový a jednofázový zkrat se jedná o fázi a, u ostatních o fázi b.

V pravé části (označené jako příspěvky větví) je nejprve vypsána elektromotorická síla E (pokud je do postiženého uzlu připojen blok) a sumární vstříkovaný proud. Pokud do uzlu není připojen blok, vypisuje se slovo GROUND a proud, který odtéká do země. Následuje výpis postižené větve, který obsahuje napětí ve vzdáleném uzlu a tok proudu ze vzdáleného uzlu a to fázové i sousledné hodnoty. U jednofázového zkratu se vypisují i hodnoty v zpětné a netočivé složce. Pak následují ostatní větve připojené do postiženého uzlu s výpisem fázových i sousledných hodnot U a I.

Program nebere v úvahu fázový posun, daný zapojením traf hvězda-trojúhelník. Při jednofázovém zkratu jsou proudy na straně hvězdy spočítány dobře, ale na straně trojúhelníka je proud v postižené fázi dvojnásobný proti správné hodnotě.

- Pro trojfázový a jednofázový zkrat je výpis doplněn o dvě sekce. První sekci tvoří příspěvky ekvivalentu soustavy v okamžiku vzniku zkratu (tzv. počáteční rázové hodnoty) a jeho struktura pro trojfázový kovový zkrat ( $Z_d=0$ ) je následující.

Prispevky ekvivalentu soustavy	Proud	[kA]	[st.]	Impedance ekvivalentu R[Ohm]	X[Ohm]
>>>IR=	1.42	-86.9	Zs	0.724	22.692
<<<IQ=	4.09	-89.1	Zl	1.600	8.000
<<IQR=	0.62	-85.4	Ze	4.160	18.346
<IQQ=	0.59	-88.4	Zuu	3.138	98.334
<IQ1 =	2.88	-90.0	Zb+Zd	0.126	20.000
<IQ2 =	1.24	-82.0	Zb+Zd	0.566	47.925
<IQ3 =	0.61	-82.0	Zb+Zd	1.731	97.692

Program bere v úvahu i změnu impedance po jednostranném odpojení a tedy časové konstanty jsou obecně různé pro oba úseky zkratu. V případě, že příslušný příspěvek nevznikne, je impedance teoreticky nekonečná a vypisují se hvězdičky. Příspěvky mají obecně různou fázi a pro kontrolu platnosti Kirchhoffova zákona je nutno sčítat komplexní hodnoty (fázory). Příspěvek IQQ má obecně nenulovou hodnotu a představuje podíl bloků, které nejsou přímo připojeny do blízkého uzlu Q (blízkým uzlem nazýváme uzel, kde je zadán zkrat). Počet příspěvků IQi odpovídá počtu bloků přímo připojených do blízkého uzlu Q.

Pro ostatní zkraty se místo Impedancí ekvivalentu vypisují impedance cesty, z kterých se dopočítává časová konstanta podle bodu 8 kap.2.5.4.

- Druhou sekci tvoří příspěvky zkrat.proudu získané z rozšířeného aktivního dvojbranu - detailní výpis veličin jednotlivých příspěvků a sumárních proudů IQ (který je vypínán vypínačem Q) a IK (sumární zkratový proud) a jeho struktura je následující:

	IR>>>	\IZKR\	<<<IQ=	<<IQR +	<IQQ+	<IQ1 +	<IQ2 +	<IQ3
I"k[kA] :	1.20	6.70	5.50	1.39	0.39	1.88	1.24	0.61
Ikm (Km) :	3.07 (1.80)	17.98 (1.90)	14.92 (1.92)	(1.82)	(1.73)	(1.98)	(1.96)	(1.95)
Ivyp(mi) :	1.01 (0.83)	4.73 (0.71)	3.75 (0.68)	(0.83)	(0.46)	(0.72)	(0.72)	(0.72)
iavyp[%] :	12	63	76	15	8	108	95	79
Ike (Ke) :	1.34 (1.11)	8.45 (1.26)	6.59 (1.20)	(1.13)	(0.86)	(1.49)	(1.44)	(1.36)
Ta [s] :	0.0444			0.0493	0.0309	0.4045	0.2696	0.1798

U jednotlivých příspěvků je výpis doplněn o příslušné časové konstanty Ta.

- závěrečný úsek obsahuje veličiny zkratového proudu pro všechny typy zkratů. Veličiny zkratového proudu. jsou vypočítané dle metodiky ČSN 33 3020 s tím rozdílem, že hodnota souměrného zkratového proudu je časově proměnná a počítaná dynamicky . Bere tedy v úvahu přechodné děje v elektrických obvodech točivých strojů, přesněji změny elektromotorických sil v sousledné složkové soustavě. Výpis je doplněn i příslušnými koeficienty Km, Ke a  $\mu$ , tak jak je definuje norma.

Veliciny zkratoveho proudu podle CSN 33 3020 tk[s] = 0.150				
	3_PH	S_L_G	D_L_G	L_L
I"k[kA] :	5.83<	7.94<	6.585	5.047
Ikm[kA],Km :	16.215,1.9673	22.088,1.9673	18.317,1.9673	14.040,1.9673
Ivyp[kA],mi :	4.694,0.8052	7.213,0.9083	5.464,0.8297	4.537,0.8989
iavyp[%] :	75	66	73	67
Ike[kA],Ke :	7.105,1.2189	11.679,1.4708	8.431,1.2804	7.311,1.4484
Ta[s] :	0.3002	0.3002	0.3002	0.3002
Pro trojfázový	Pro jednofázový	Pro dvoufázový	Pro dvoufázový	

V případě, že dojde k chybě při výpočtu impedance  $Z_{K0}$  vypisuje se pro jednofázový zkrat nulová hodnota  $I''_k$ . Program hlásí (v souboru AKCE.HLA): Doslo k chybe pri vypoctu triangularizovene adm.matice - deleni nulou v uzlu #. Větve připojené do uzlu s uživatelským číslem # je nutno vyřadit ze souboru NESYM.DAT.



## 2.6 Automatická tvorba hromadných scénářů programem SCENARE

Náplní dynamické simulace na počítačovém modelu je zadávání různých poruch a sledování odezvy systému. Poruchy se zadávají pomocí **scénáře**, který je uložen ve **vstupním souboru** SCENAR.DAT. Pomocný program SCENARE umožňuje vytvořit tento soubor pro hromadné scénáře (pro všechny objekty daného typu) pro následující poruchy:

1. výpadky větví
2. výpadky bloků (generátorů)
3. trojpolové zkratky v určeném místě větve a s určenou dobou vypnutí.

Specifikace poruchy se provede ve **vstupním souboru** SCENARE.DAT jehož struktura je na následujícím výpisu:

Soubor SCENARE.DAT pro zadání parametru pomocného programu SCENARE

Typ: Určuje typ poruchy\*, Volba 1- zápis do trénovací množiny 0-neprovádí se

3 \_\_\_\_\_ 1 \_\_\_\_\_

Tvych [s] čas zadání poruchy a zápisu výchozího stavu

1 \_\_\_\_\_

Tsim [s] doba simulace poruchy - čas mezi zápisem výchozího a výsledného stavu

10 \_\_\_\_\_

Tzkrat [s] doba vypnutí zkratu (pro Typ=3)

0.15 \_\_\_\_\_

Vzdal [%] procentní vzdálenost místa zkratu od počátečního uzlu (pro Typ=3)

0 \_\_\_\_\_

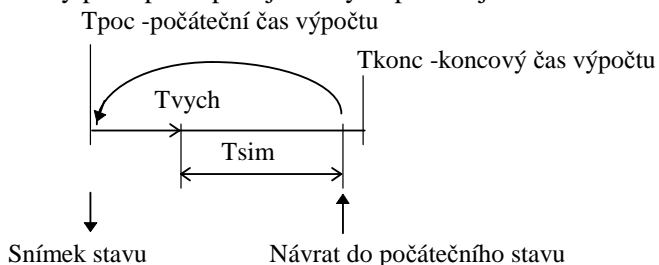
\*Typ=1 pro výpadky větví

=2 pro výpadky bloku

=3 pro zkratky

První parametr Typ určuje číselný kód poruchy (viz legenda). Druhý parametr při volbě 1 zadává zápis do trénovacího souboru TREN.BIN, volbou 0 se zápis neprovádí.

Další dva parametry určují časový průběh poruchy. Tvych definuje čas zadání poruchy ve scénáři a Tsim pak dobu trvání simulace poruchy. Po skončení této doby se ve scénáři zadá návrat do počátku výpočtu a celý postup se opakuje do vyčerpání objektů hromadného scénáře. Situace je znázorněna na následujícím:



**Obr. 3 časový průběh výpočtu při hromadném scénáři**

Počáteční čas výpočtu je při volbě parametru Ipoc=0 a 1 ve **vstupním souboru** RIZENI.DAT automaticky nula. Koncový čas výpočtu se zadává opět ve **vstupním souboru** RIZENI.DAT. Na počátku výpočtu se provede uložení stavu (hodnoty stavových proměnných). V čase Tvych se zadává porucha. Simulace poruchy probíhá po dobu Tsim. Pak se provede alternativní zápis do trénovacího souboru TREN.BIN (při Volba=1) a návrat do počátečního času a celý postup se opakuje. Po poslední poruše hromadného scénáře výpočet doběhne do času Tkonc a skončí. Pro úspěšné provedení hromadného scénáře vytvořeného pomocným programem SCENARE je tedy nutné aby  $Tvych > Tpod$  a  $Tvych + Tsim < Tkonc$ .

Poslední dva parametry se týkají zadání zkratu (pro Typ=3). Tzkrat udává dobu trvání zkratu - rozdíl času mezi zadáním poruchy Tvych a dobou vypnutí postižené **větvě**. Druhý parametr udává procentní vzdálenost místa zkratu od počátečního **uzlu větve** (počáteční a koncový **uzel** je definován ve **vstupním souboru** VET.DAT).

## 2.7 Tvorba automatik a logik z šablon programem SABLONY

Pomocný program SABLONY (dostupný příkazem Tools/Automatics and Logics Templates) umožňuje doplnit vstupní soubory AUTOMAT.DAT a LOGIC.DAT novými automatikami a logikami na základě připravených šablon. Tyto šablony jsou uloženy v souborech AUTOMAT.SAB a LOGIC.SAB, které jsou v podadresáři SKLAD pracovního adresáře MODESu.

Údržba spočívá v tom, že jedno řádky jednou odladěných automatik/logik zkopíruje do souborů AUTOMAT/LOGIC.SAB a zamění konkrétní jména větví, uzlů a větví za klíčová slova VETEV, UZEL a BLOK. Na konec řádku lze dopsat komentář začínající znakem !. Komentář obsahuje informace, které jsou zobrazovány programem SABLONY a slouží k orientaci uživatele při tvorbě nových automatik/logik. Příklad záznamu šablony automatiky je na následujícím výpisu“

```
1 'azr_13' 0.02 '/U/' 'UZEL' ' ' ' ' 'LIMT' 'LT' 3 0.0.0 0.0 .1 1000 'BRAN' 'VETEV' 1
!UZEL-VETEV, podpetove rele zapojuje zaskok jednoduchy AZR [MOTOR]
```

Bezprostředně za vykřičníkem před čárkou je uveden typ šablony, kde jsou uvedena klíčová slova. V daném příkladu program SABLONY po stisknutí tlačítka Prepare objects nabídne uživateli konkrétní jména uzlů a větví příslušející danému případu. uživateli. Uživatel si vybere ze seznamu objekt a potvrdí náhradu klíčového slova jménem vybraného objektu tlačítkem Select object. Záměnu je vidět ve spodním řádku v rámečku Specification. Tlačítkem Add to Table se daný záznam přidá do tabulky automatik.

Další položkou v komentáři za čárkou je stručný popis automatiky/logiky, který umožňuje uživateli výběr ze seznamu šablon. V hranaté závorce je jméno projektu, kde již byla daná automatika/logika použita.

V případě, že automatika/logika provádí hromadný nebo vícenásobný zásah, je nutno použít tzv. makra. Příklad takové šablony automatiky je na následujícím výpisu:

```
1 'AktAZR' 0.1 '#B' 'VETEV' ' ' ' ' 'LIMT' 'EQ' 0 0.0.0 0.04 1.1 1 $(AktAZR:)
!VETEV-AUTO, aktivuje "AZR" po vypnutí vetve; aktivuje 'checkf' 'Vypnut' [MOTOR]
```

Pokud program narazí na znak \$ přesměruje čtení do souboru MAKRA.DAT, kde načte informace o zásahu hromadném (zásah jednoho typu na více objektů) nebo vícenásobném (více zásahů různého typu na jedno působení automatiky/logiky).

Soubor MAKRA.DAT může být buď globální - dostupný všem projektům (uložený v podadresáři GLOBAL.DAT) nebo lokální - dostupný pouze danému projektu (uložený v podadresáři VST). Program MODES prohledá nejprve lokální soubor a pokud v něm dané makro nenajde, prohledává globální tabulku maker. V obou souborech tedy mohou být stejnojmenná makra, ale lokální makro má prioritu.

V komentáři za středníkem jsou uvedeny související automatiky/logiky, které jsou definovány v makru. Tyto automatiky/logiky je nutno také přidat do příslušných tabulek. Informace o těchto souvisejících automatikách/logikách jsou v textovém okně Related to: ve specifikaci šablony.

Některé automatiky pracují v tandemu (např. jedna spouští nebo blokuje druhou). Informace o tom jsou v komentáři za znakem > a promítá se do textového okna Link to:. Opět je nutno pro správnou funkci přidat do tabulky obě šablony.

Stisknutím tlačítka OK se zaktualizuje vstupní soubor AUTOMAT/LOGIC.DAT. Ve stavovém řádku programu SABLONY jsou nápovědné informace.

### 3. PŘÍDAVNÉ PROGRAMY

Ve verzi 2.2/2 byly některé funkce vyjmuty z programu MODES a svěřeny tzv. přídatným programům. Jedná se o méně často používané funkce - výpočet středních kvadratických odchylek a veličin sekundární regulace P a f. Hlavním důvodem bylo zeštíhlení programu a zjednodušení jeho obsluhy. Přídatné programy jsou volitelnou součástí dodávky balíku programů MODES a vlastní jádro programu pro ně pouze vytváří příslušné **vstupní soubory**. V případech uvedených výše jsou to **soubory** BLOK\_ANA.SRI a LFC\_ANA.SRI, sloužící jako vstup pro přídatný program ODCHYLKY.

Nové funkce (harmonická analýza) jsou již realizovány na tomto principu pomocí **souboru** HARM\_ANA.SRI a přídatného programu HARM\_ANA.

#### 3.1 Nástroj ODCHYLKY

##### 3.1.1 Výpočet středních kvadratických odchylek

Pro určité proměnné, které by měly být v **ustáleném stavu** nulové, je možné zadat výpočet střední kvadratické odchylky podle vztahu:

$$\sigma = \sqrt{\left\{ \left[ \int_{t_{\text{POCAT}}}^{t_{\text{KONC}}} (x^2 dt) \right] / (t_{\text{KONC}} - t_{\text{POCAT}}) \right\}}$$

Kde za  $x$  je možno dosadit následující **proměnné**:

- skluz generátoru
- regulační odchylku regulátoru jalového výkonu
- regulační odchylku regulátoru buzení
- regulační odchylku regulátoru turbíny
- regulační odchylku regulátoru kotle

Zadání **symbolu proměnné** a **bloku** se provede ve **vstupním souboru** ANAL.DAT. Definované proměnné se ukládají do **výstupního souboru** BLOK\_ANA.SRI, který vznikne automaticky v adresáři VYST a slouží jako vstup pro přídatný program ODCHYLKY. Tento program provede vlastní výpočet středních kvadratických odchylek a uloží je do **výstupního souboru** BLOK\_ANA.SRO.

Hodnotu stř. kvadratické odchylky regulačních odchylek je možné využít pro porovnání dvou nastavení parametrů regulátoru. Kvalitnější regulační proces s menším přeregulováním a s rychlejším tlumením má odchylku menší.

##### 3.1.2 Analýza veličin sekundární regulace P a f

Při zadání parametru Ana4=1 ve **vstupním souboru** ANAL.DAT program vytvoří pomocný **výstupní soubor** LFC\_ANA.SRI, který slouží jako vstup pro přídatný program ODCHYLKY, který je volitelnou součástí dodávky. Tento program pak dopočítá veličiny charakterizující sekundární regulační proces a uloží je do **výstupního souboru** LFC\_ANA.SRO.



Je možné např. určit velikost tzv. „regulační energie“, tj. hodnoty energie, kterou daná **oblast** importuje/exportuje v daném časovém intervalu a tak usuzovat na kvalitu regulace P a f.

### 3.2 Harmonická analýza časových průběhů programem HARM\_ANA

Program MODES ve verzi 2.2/2 umožňuje pomocí **zásahů** 'EXCH'/'TURH' přidat harmonický signál (sinusového průběhu) o definované amplitudě a frekvenci do součtového členu regulátoru buzení/pohonu. Tím se do ES zavádějí vynucené kyvy.

Ve **vstupním souboru** ANAL.DAT se v úseku harmonické analýzy dají definovat **proměnné** vybraných **bloků**, jejichž časový průběh se během výpočtu ukládá do pomocného **výstupního souboru** HARM\_ANA.SRI. Zároveň se do tohoto souboru tisknou i údaje o velikosti amplitudy a frekvence přídavného signálu v okamžiku zadání **zásahu**. Tento soubor se pak použije jako vstup přídavného programu HARM\_ANA, který je volitelnou součástí dodávky. Tento program pak vyhodnotí poměr amplitud indukovaných kyvů vybraných **proměnných** a amplitudy budícího harmonického signálu (uvádí se v seznamu **proměnných** jako první) a rozdíl jejich fází (fázový posuv). Tyto hodnoty se vypíší v tabulkové formě do **výstupního souboru** HARM\_ANA.SRO a mohou se použít pro konstrukci frekvenčních charakteristik.



Uvedený postup má praktické použití při návrhu a optimalizaci parametrů systémových stabilizátorů, které slouží k tlumení tzv. systémových kyvů<sup>1</sup>.

## 4. ZÁVĚR

Tato příručka poskytuje stručné základní informace, nutné pro práci s pomocnými a přídavnými programy balíku MODES. Detailnější informace obdrží uživatel při instalaci programu, pomocí telefonických dotazů (hot line), případně doplňkovým zaškolením.

## LITERATURA

- [1] Karel Máslo: Popis modelování přechodných dějů v elektrizační soustavě,  
Použití modelů v programu MODES
- [3] Karel Máslo: Používání programu MODES pro výpočet přechodných dějů v elektrizační soustavě,  
Uživatelská příručka programu MODES
- [3] Karel Máslo: Průvodce použitím programu MODES při analýze dynamického chování ES,  
Metodický průvodce MODES
- [4] ČSN 33 3020: Výpočet poměrů při zkratech v trojfázové ES, Úřad pro normalizaci a měření,  
Praha 1992
- [5] P.M. Anderson: Power system protection, IEEE press series on power engineering,  
McGraw-Hill, 1999
- [6] ČSN 33 3020: Výpočet poměrů při zkratech v trojfázové ES, Úřad pro normalizaci a měření,  
Praha 1987

<sup>1</sup> Podrobné postup najde uživatel v Průvodci [2]

## 5. Příloha -Způsob výpočty veličin zkratového proudu

Materiál ukazuje metodiku výpočtu nárazového a vypínacího zkratového proudu a ekvivalentního oteplovacího proudu v programech MODES a ZKRATY. Vychází se z toho, že zkratový proud se skládá ze souměrné střídavé složky o jmenovitém frekvenci  $\omega=2\pi f_n$  a stejnosměrné složky zanikající s časovou konstantou  $T_a$ :

$$i_K(t_{rel}) = \sqrt{2}[-I_{zkr}(t_{rel})\cos\omega t_{rel} + I_k'' e^{-t_{rel}/T_a}] \quad t_{rel}=t-t_0 \quad [1]$$

$t_{rel}$  je relativní čas od vzniku zkratu v čase  $t_0$ . Střídavá složka má proměnnou amplitudu  $I_{zkr}(t_{rel})$ , která je výsledkem síťového výpočtu na dynamickém modelu ES. Počáteční hodnota stejnosměrné složky odpovídá počátečnímu rázovému proudu:

$$I_k'' = I_{zkr}(0) \quad \mu(t_{rel}) = I_{zkr}(t_{rel})/I_k'' \quad [2]$$

### 5.1 Nárazový zkratový proud

Nárazový zkratový proud  $I_{km}$  se určí jako součet maxim příspěvků, což je v souladu s doporučením ČSN 33 3020 článku 4.3.3. Maximum příspěvku se spočítá jako extrém časového průběhu  $i_K$  podle definice:

$$I_{km} = \sqrt{2} * K_m * I_k'' = \max\{i_K(t_{rel})\} \text{ pro } t_{rel} \leq 0, t_k > 0 \quad [3]$$

Program určí hodnotu  $I_{km}$  za předpokladu, že k extrému funkce [1] dojde v čase  $t_{max}$ , přičemž tento čas se určí přibližným řešením rovnice  $dI_{km}/dt_{rel} = 0$  v okolí času 0.01s (polovina periody při jmenovité frekvenci 50 Hz). Při výpočtu maximální hodnoty  $I_{km}$  se bere v úvahu i pokles střídavé složky zkratového proudu pomocí koeficientu  $\mu$ . Pro součinitel  $K_m$  pak platí:

$$K_m = \mu(0.01)\cos\omega t_{max} + e^{-t_{max}/T_a} \quad t_{max} \approx 0.01 - x/(\omega T_a - x/\omega T_a)/\omega \quad [4]$$

$\mu(0.01)$  je podíl střídavé složky pro čas 0.01 s  $\geq t_{rel}$  (obvykle během prvního integračního kroku po zkratu) a počátečního zkratového proudu. Pro porovnání uvádíme zjednodušený vztah 42a pro tzv. příspěvek ze soustavy podle normy /1/:

$$K_m = 1 + e^{-0.01/T_a} \quad [5]$$

. Hodnoty  $t_{max}$  pro některé časové konstanty a  $K_m$  vypočítané podle vztahů [4] (pro  $\mu=1$ ) a [5] jsou v následující tabulce:

$T_a$	0.006	0.02	0.1	0.5
$t_{max}$	0.00966	0.0096878	0.009908	0.00998
$K_m = \cos\omega t_{max} + e^{-t_{max}/T_a}$	1.194	1.611	1.9	1.98
$K_m = 1 + e^{-0.01/T_a}$	1.188	1.606	1.9	1.98

Pokud je pro okamžitou hodnotu některého z příspěvků použita aproximace pomocí hyperbolického sinu/kosinu použije se pro výpočet maxima přímo definiční vztah [3], který se kontroluje s krokem 1ms (20 vzorků za periodu).

### 5.2 Vypínací zkratový proud

Souměrný vypínací proud je dán vztahem:

$$I_{vyp} = I_{zkr}(t_k) = \mu_{vyp} I_k'' \quad [6]$$

kde  $t_k$  je doba trvání zkratu (dána scénářem zadání zkratu a jeho vypnutí).  $I_{vyp}$  je výsledkem síťového výpočtu na dynamickém modelu ES. Součinitel  $\mu_{vyp}$  je tedy určen na základě přesného řešení příslušných diferenciálních rovnic synchronních, případně i asynchronních strojů včetně vlivu buzení.

Procentní hodnota stejnosměrné složky  $I_{avyp}$  vypínacího zkratového proudu je určena definičním vztahem:

$$i_{avyp} = 100 * I_{avyp} / I_{vyp} / \sqrt{2} \quad [7]$$

a pro [1] je dána vztahem:

$$i_{avyp} = 100(e^{-t_k/T_a} I_k'' / I_{vyp}) = 100(e^{-t_k/T_a} / \mu_{vyp}) \quad [8]$$

### 5.3 Ekvivalentní oteplovací proud

Ekvivalentní oteplovací proud je dán vztahem:

$$I_{ke} = \sqrt{(Q / t_k)} = K_e * I_k \quad [9]$$

kde Jouleův integrál  $Q$  na časovém intervalu  $<0, t_k>$  je v programu řešen v závislosti na časově závislém součiniteli  $\mu(t_{rel})$

$$Q = \int_0^{t_k} i_k(t_{rel})^2 dt_{rel} \quad [10]$$

Pokud se uvaží časový průběh zkratového proudu podle [1] je Jouleův integrál  $Q$  roven:

$$2 I_k^2 \int_0^{t_k} [\mu(t_{rel})^2 (1 + \cos 2\omega t_{rel}) / 2 - 2\mu(t_{rel}) e^{-t_{rel}/T_a} \cos \omega t_{rel} + e^{-2t_{rel}/T_a}] dt_{rel} \quad [11]$$

Jelikož v programech MODES/ZKRATY není čas spojitý, ale se vzorkováním v intervalu integračního kroku  $\Delta t_i$ , provádí se výpočet přírůstku Jouleova integrálu  $\Delta Q$  pro každý integrační interval  $i < t_i, t_i + \Delta t_i >$  a pro příspěvek  $K$  podle vztahu (při zanedbání harmonických členů<sup>1</sup>):

$$dQ_{iK} = I_k^2 [\Delta t_i \mu_i^2 + T_{ai} e^{-2t_0/T_{ai}} (1 - e^{-2\Delta t_i / T_{ai}})] \quad K = R, RQ, QQ, Q1, \dots, Qn \quad [12]$$

kde  $\mu_i$  je střední hodnota na časovém intervalu  $i$ . Sumární přírůstek se pak určí vztahem:

$$dQ = \sum_{i=1, N} \left[ k_i^2 dQ_i + \sum_{j=i+1, N} 2k_i k_j dq_{ij} \right] \quad dq_{ij} = \Delta t_i \mu_i \mu_j + T_{ij} e^{-2t_0/T_{ij}} (1 - e^{-2\Delta t_i / T_{ij}}) \quad T_{ij} = T_{ai} T_{aj} / (T_{ai} + T_{aj}) \quad k_i = I_{ki} / I_{ki} \quad [13]$$

což je zobecněním vztahu pro Jouleův integrál dvou příspěvků (uvedený jako (63) v ČSN33 3020<sup>2</sup>) na obecně  $N$  příspěvků, kde  $N$  je počet přímo připojených bloků  $n$  zvýšený o  $2/3$  pro proudy IQ/IZKR.

Pokud je pro okamžitou hodnotu některého z příspěvků použita aproximace pomocí hyperbolického sinu/kosinu použije se pro výpočet Jouleova integrálu definiční vztah [10], přičemž se integrál nahradí sumací a diferenciál času  $dt_{rel}$  krokem  $1ms$ .

<sup>1</sup> Doba trvání zkratu je obvykle několik period střídavého proudu, takže kladné a záporné plochy kosinusovky se eliminují

<sup>2</sup> Je třeba uvést, že v normě je chyba - v posledním členu má být  $e^{-t_k/T_{12}}$  a ne  $e^{-2t_k/T_{12}}$

## OBSAH

	PŘEDMLUVA .....	1
	TISKOVÁ KONVENCE .....	3
	DEFINICE POUŽITÝCH TERMÍNŮ .....	3
<b>1.</b>	<b>SOUVISLOST MEZI POMOCNÝMI PROGRAMY .....</b>	<b>4</b>
<b>2.</b>	<b>POMOCNÉ PROGRAMY .....</b>	<b>5</b>
2.1	TRANSFORMACE SOUBORŮ RAW PROGRAMEM PSSEDATA .....	5
2.2	REDUKCE POČTU UZLŮ PROGRAMEM EKVIVAL .....	7
2.3	VÝPOČET USTÁLENÉHO STAVU (US) POMOCÍ PROGRAMU UST .....	9
2.3.1	Spuštění programu UST .....	9
2.3.2	Ladění ustáleného stavu .....	9
2.4	VÝPOČET ČASOVÉHO PRŮBĚHU A CHAR.HODNOT ZKRATOVÉHO PROUDU ZKRATY .....	12
2.4.1	Účel programu .....	12
2.4.2	Instalace programu .....	12
2.4.3	Předpoklady fungování programu .....	12
2.4.4	Metodika výpočtu .....	13
2.4.5	Spouštění programu .....	14
2.4.6	Výsledky zkratového výpočtu .....	14
2.5	AUTOMATICKÁ TVORBA HROMADNÝCH SCÉNÁŘŮ PROGRAMEM SCENARE .....	17
2.6	TVORBA AUTOMATIK A LOGIK Z ŠABLON PROGRAMEM SABLONY .....	18
<b>3.</b>	<b>PŘÍDAVNÉ PROGRAMY .....</b>	<b>19</b>
3.1	NÁSTROJ ODCHYLKY .....	19
3.1.1	Výpočet středních kvadratických odchylek .....	19
3.1.2	Analýza veličin sekundární regulace P a f .....	19
3.2	HARMONICKÁ ANALÝZA ČASOVÝCH PRŮBĚHŮ PROGRAMEM HARM_ANA .....	20
<b>4.</b>	<b>ZÁVĚR .....</b>	<b>20</b>
	LITERATURA .....	20
<b>5.</b>	<b>PŘÍLOHA -ZPŮSOB VÝPOČTY VELIČIN ZKRATOVÉHO PROUDU .....</b>	<b>21</b>
5.1	NÁRAZOVÝ ZKRATOVÝ PROUD .....	21
5.2	VYPÍNAČÍ ZKRATOVÝ PROUD .....	21
5.3	EKVIVALENTNÍ OTEPLOVACÍ PROUD .....	22
	OBSAH .....	23