

## 1. ÚVOD

Tento průvodce slouží uživatelům programu MODES verze 2.2/4 pro seznámení s možnostmi a pracovními postupy tohoto balíku programů pro analýzu elektrizační soustavy. Tvoří doplněk k základní dokumentaci programu tj. Popisu modelování [1] a Uživatelské příručky [2].

Průvodce má dva díly. První díl slouží začínajícím uživatelům a zájemcům o využívání MODESu. Seznamuje uživatele se základními koncepty práce s programem (projekty, případy, variace), způsobem modelování (základní a modifikované databáze, globální a lokální katalogy, scénář). Druhý díl slouží zkušenějším uživatelům pro seznámení s koncepty automatik a logik.

Průvodce je členěn do oddílů podle jednotlivých okruhů, se kterými se setkáme při analýze dynamického chování elektrizační soustavy (ES). Jedná se o tyto okruhy problémů:

- 1) příprava základního modelu elektrizační soustavy
- 2) krátkodobá dynamika - řeší přechodovou stabilitu synchronního stroje<sup>1</sup> po poruchách typu zkratů
- 3) střednědobá dynamika - řeší otázky frekvenčního kolapsu po poruchách typu výpadku bloků
- 4) dlouhodobá dynamika - řeší činnost sekundární regulace P/f a otázky napěťového kolapsu
- 5) přechodné děje ve vlastní spotřebě - řeší samonajíždění a spouštění asynchronních pohonů
- 6) ostrovní režimy - řeší otázky přechodu částí ES do ostrovního provozu a jejich resynchronizace

Průvodce tak pokrývá většinu problémů analýzy ES, které je třeba řešit při:

- výuce a studiu přechodných dějů
  - návrhu a testování ochranných a řídicích prvků
  - provozních studiích (např. tvorba havarijních a obnovovacích plánů)
- a shrnuje tak dlouholeté zkušenosti autora v oblasti pedagogické (ČVUT Praha), výzkumné (EGÚ Praha) a praktické (v Oddělení systémových analýz Divize přenosové soustavy ČEZ, a.s.).

Doporučuje se probírat oddíly postupně, neboť jednotlivé rysy programu jsou popsány podrobně (ve formě metodických postupů případně zobrazení dialogů nebo výpisů vstupních souborů) pouze pokud se na ně narazí poprvé. Doporučuje se podrobně seznámit s používanými pojmy, které jsou definovány v kapitole 1.2 **Chyba! Neznámý argument přepínače.**

Průvodce metodicky seznamuje uživatele s jednotlivými kroky analýzy dynamického chování ES a zároveň s jednotlivými koncepty programu MODES, které tyto kroky realizují. Všechna vstupní data jsou připravena ve formě tzv. **projektů** a **případů**. Průvodce tedy neřeší otázku sběru a přípravy vstupních dat, nicméně ukazuje, která vstupní data jsou pro řešení potřebná.

Pro modelovanou síť byl zvolen testovací případ úkolu CIGRE<sup>2</sup> 38-02-08 Dlouhodobá dynamika [3], který byl v případě potřeby upraven. Pro **parametry** dynamických modelů byly použity **katalogy** typových parametrů, které tvoří standardní dodávku balíku programů MODES. **Variace scénářů** se používají pro simulaci poruch a činnosti obsluhy, které je potřeba uskutečnit v určitém čase<sup>3</sup>. Předdefinované **automatiky** a **logik** se používají pro simulaci reálných ochranných a řídicích prvků, které určitou činnost (**zásah**) vykonávají automaticky při splnění definované podmínky. Přehled **zásahů** je v Příručce [2] a popis fungování **automatik** a **logik** je v Popisu [1].

Těžiště práce s programem představuje jednak příprava vstupních dat (ladění modelu) a jednak rozbor a vyhodnocení výsledků. Průvodce je zaměřen zejména na přípravu data a vyhodnocení výsledků se dotýká okrajově. Při úpravě vstupních dat je možno zvolit jeden ze dvou režimů: textový nebo dialogový (zaškrtnutím volby **Dialogy/Tabulky/Databáze** v menu **Modifikovat**).

Dialogový režim podporuje tzv. **variace** a **modifikace** vstupních souborů typu **formulářů** a **databází**. Tím umožňuje tvorbu nových **případů**. Tento režim není dostupný v DEMO verzi, takže lze pouze **případy** otevírat.

Textový režim pracuje se vstupními soubory přímo pomocí textového editoru. Práce je rychlejší, ale vyžaduje znalost struktury vstupních datových souborů (viz konvence a popis v kapitole 3. Druhy vstupních souborů v Příručce [2]).

<sup>1</sup> přechodovou („transient“) stabilitou synchronního stroje rozumíme schopnost udržet se v synchronním chodu

<sup>2</sup> „Conférence Internationale des Grands Réseaux Electriques“ se zabývá pokrokem ve výrobě a přenosu elektřiny

<sup>3</sup> jinak se pro simulaci zásahů obsluhy uživatelem během výpočtu používají horké klíče F1 - F8

## 1.1. Tisková konvence

V Průvodci jsou použity následující typografické prostředky pro usnadnění orientace čtenáře v textu.

### 1.1.1. Typy písma

Bezpatkové písmo pro výpisy souborů a pro jména objektů<sup>1</sup>

**Tučné bezpatkové písmo** pro příkazy menu v uživatelském rozhraní

*Kurzíva* pro názvy dialogových oken a popisy jejich prvků

**Tučná kurzíva** pro názvy tlačítek v dialogových oknech

**Tučné písmo** pro pojmy definované v kapitole 1.2 **Chyba! Neznámý argument přepínače.**

‘KLIC’ v apostrofech jsou klíčová slova<sup>2</sup> programu

JMENO . PRI velké neproporcionální písmo pro názvy vstupních a výstupních souborů programu

### 1.1.2. Názvy titulků

Pro rozlišení jednotlivých obrázků a výpisů souborů se používají tyto titulky:

Ana. pro výpisy výstupních souborů s analýzou

Hla. pro výpisy výstupních souborů s hlášením programu během výpočtu

Obr. pro znázornění grafů na obrazovce během výpočtu programu

Sch. pro jednopólové schémata sítě s naznačením simulovaných poruch

Výp. pro výpisy vstupních souborů programu.

Dial. Pro zobrazení dialogových oken

### 1.1.3. Grafické zvýraznění

XXXXX pro zvýraznění vstupních souborů  
xxxxxxx

XXXXXXX pro zvýraznění výstupních souborů  
xxxxxx

☞ Důležité body:

- Ukazuje koncepty a postupy programu MODES při dynamické simulaci ES

⊕ Potenciální problémy:

- ♦ Upozorňuje na rysy programu, které mohou způsobit problémy

Nové rysy zavedené ve verzi 2.2/3 jsou vyznačeny čarou na vnějším okraji .

Dvojitou čarou jsou vyznačeny zkrácené postupy, kdy se používají předdefinované **případy**.

☞ uvozuje jednotlivé kroky určitého postupu

☞ uvozuje textový režim úpravy vstupních souborů v uživatelském rozhraní

☒ uvozuje dialogový režim úpravy vstupních souborů (při zaškrtnutí volby **Dialogy/Tabulky/Databáze** v menu **Modifikovat**). (

### 1.1.4. Nadpisy

#### **NADPIS ODDÍLU**

#### **Nadpis kapitoly**

### 1.1.5. Popis příkazů uživatelského rozhraní

Výraz **Modifikovat|Řízení** je zkrácený zápis pro: z menu **Modifikovat** provést příkaz **Řízení**

<sup>1</sup> tj. jmen uzlů, bloků, větví, profilů, oblastí a automatik

<sup>2</sup> názvy symbolů proměnných, zásahů, knihovních modelů a typů měřících článků automatik

## 1.2. Použité pojmy

V Průvodci jsou použity následující termíny pro práci s programem MODES

<b>archiv</b>	soubor obsahující informace o <b>projektech</b> , <b>případech</b> a editacích
<b>automatika</b>	model, který na základě měření průběhu vybrané <b>proměnné</b> provede <b>zásah</b> na vybraný <b>objekt</b>
<b>blok</b>	<b>objekt</b> skládající se ze synchronního nebo asynchronního soustrojí
<b>databáze</b>	vstupní soubor se záznamy s definicí dynamických <b>modelů objektů</b>
<b>formulář</b>	vstupní soubor s definicí funkce programu
<b>grafika</b>	zprostředkuje styk uživatele s programem po jeho spuštění
<b>hlášení</b>	zprávy programu ukládané do výstupního souborů s příponou HLA
<b>inicializace</b>	proces načítání vstupních dat a výpočet počátečních podmínek modelů
<b>katalog parametrů</b>	obsahuje sady typových parametrů pro jednotlivé dynamické <b>modely</b>
<b>knihovna modelů</b>	obsahuje hotové <b>modely</b> prvků elektrizační soustavy
<b>logika</b>	nadstavba <b>automatik</b> , která na základě vyhodnocení splnění podmínek logického součtu/součinu provede hromadný a/nebo vícenásobný <b>zásah</b>
<b>modifikace</b>	pozměněná <b>databáze modelů</b> , která je dostupná pouze danému <b>případu</b> .
<b>makro</b>	umožňuje vícenásobné použití zápisu <b>zásahu</b> a hromadné/násobné <b>zásahy</b>
<b>návrat</b>	postup umožňující zpětný skok v simulaci do předtím uloženého stavu
<b>objekt</b>	prvek modelovaný jako pojmenovaná entita - <b>blok</b> , <b>uzel</b> , <b>větev</b> , <b>oblast</b> , <b>profil</b> , <b>automatik</b> nebo <b>logik</b>
<b>oblast</b>	množina <b>uzlů</b> definovaných ve vstupním souboru UST.DAT
<b>projekt</b>	souhrn vstupních a výstupních dat, který byl uživatelem specifikován a uložen příkazem <b>Projekt   Save Projekt As</b>
<b>profil</b>	množina větví definovaných ve vstup. souborech UST.DAT a KONT.DAT
<b>případ</b>	definuje sadu vstupních souborů jednotlivého výpočtu, který byl uživatelem specifikován a uložen příkazem <b>Případ   Save Případ As</b>
<b>proměnná</b>	číselná veličina definovaná <b>symbolem</b> a jménem <b>objektu</b>
<b>přídavné automatiky</b>	autonomní funkce regulátorů buzení a turbín
<b>scénář</b>	předem definovaný časový sled <b>zásahů</b>
<b>symbol</b>	čtyřmístný alfanumerický řetězec definující typ <b>proměnné</b>
<b>události</b>	množina <b>zásahů</b> a činností <b>přídavných automatik</b>
<b>uzel</b>	modeluje bod, kde se stýkají <b>větvě</b> , jsou připojeny <b>zátěže</b> a <b>bloky</b>
<b>uživatelský soubor</b>	slouží pro ukládání časových průběhů <b>proměnných</b>
<b>variace</b>	<b>formulář</b> změněný editací, je identifikován pomocí přípony
<b>větev</b>	modeluje vedení, trať a spínače
<b>zásah</b>	simuluje změny, zadávané <b>scénářem</b> , horkými klávesami F1-F8 v <b>grafice</b> během výpočtu nebo je výsledkem činnosti <b>automatiky</b> a <b>logik</b>
<b>zátěž</b>	modeluje odběr činného a jalového výkonu v <b>uzlu</b>
Pojmy týkající se práce s uživatelským rozhraním MODMAN	
<b>bublínková nápověda</b>	stručná informace - objeví se při zastavení kurzoru nad prvkem <b>dialogu</b>
<b>kliknutí</b>	krátké stlačení levého tlačítka myši
<b>dialog</b>	režim zadávání vstupních dat pomocí dialogových oken
<b>kontextová nápověda</b>	podrobnější informace o <b>příkazu</b> dostupné po stisknutí klávesy F1
<b>menu</b>	nabídka funkcí programu, po kliknutí na položku menu se rozbalí <b>příkazy</b>
<b>příkaz</b>	položka padacího <b>menu</b> provádějící určitou činnost
<b>řádková nápověda</b>	stručná informace o <b>příkazu</b> na stavovém řádku
<b>seznam</b>	prvek <b>dialogu</b> zobrazující množinu údajů
<b>textové pole</b>	prvek <b>dialogu</b> pro vstup textu z klávesnice
<b>tlačítko</b>	po kliknutí provede určitou operaci
<b>volba</b>	výběr z několika možností pomocí přepínače („Radio buton“)
<b>zatrhnutí</b>	výběr ze dvou možností pomocí přepínače („Kontrola box“)

### 1.3. Práce v uživatelském rozhraní MODMAN

MODMAN je součástí standardní dodávky balíku programu MODES, které plní tyto funkce:

- umožňuje práci s **projekty** a **případy** a obsluhuje soubory a prostředky (Menu **Projekt** a **Případ**)
- provádí úpravy a editaci vstupních souborů (Menu **Modifikovat**)
- prohlíží standardní a uživatelské výstupní soubory (Menu **Hlášení**, **Analýza** a **Výsledky**)
- spouští program MODES, pomocné a přídavné programy (Menu **Spust'** a **Nástroje**)
- umožňuje **kontextovou** a **bublínkovou nápovědu**

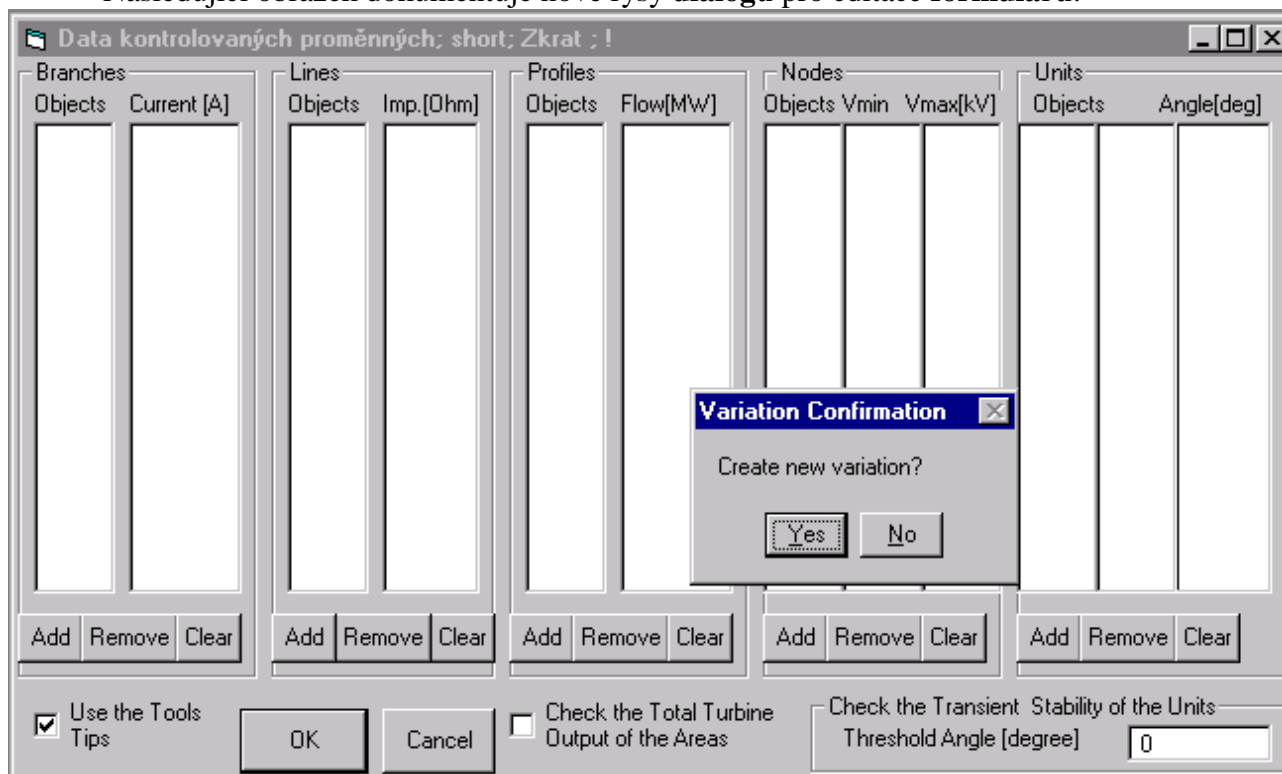
Uživatelské rozhraní se spustí kliknutím na ikonu MODMAN. Otevření projektu se provádí příkazem **Projekt| Otevřít**. Je možné zadat **projekt** pomocí jeho jména v seznamu **projektů** (doporučený režim) nebo volbou adresáře (pouze pro kompatibilitu s předchozími verzemi MODMAN). Volba adresáře se provede po zatržení *Apply ...Directory*. Pak se zvolí disková jednotka a adresář (způsobem standardním pro prostředí WINDOWS).

Editace vstupních souborů se provádí následujícími příkazy z menu **Modifikovat**:

Formuláře *.DAT	Tabulky *.DAT	Databáze modelů *.DTB	Katalogy *.CAT
Řízení⇒ RIZENI Analýza⇒ ANAL Kontrola ⇒ KONT Scénář⇒ SCENAR Grafika⇒ VYSTUP Už. soubory⇒ SOUBORY	Uzly ⇒ UST Větvě ⇒ VET řetězce pro <b>godely</b> Bloků ⇒ Automatiky⇒ AUTOMAT Regulace P/f ⇒ AUTSEK Stabilizát. a regulát.⇒ STABIL Logiky ⇒ LOGIKY	Modely uzlů ⇒ UZLY Modely sítě⇒SIT Modely Bloků ⇒ BLOK	bloků⇒TYP_BLOK sítě⇒TYP_SIT symbolů⇒SYMBOL

**Formuláře/databáze** lze upravovat ve dvou režimech v závislosti na volbě v padacím menu. Při zaškrtnutí volby **Dialogy/Tabulky/Databáze** je to dialogový režim pomocí dialogových oken, jinak textový režim pomocí vestavěného textového editoru,.

Následující obrázek dokumentuje nové rysy **dialogů** pro editace **formulářů**:



V hlavičce **dialogu** je český popis jeho funkce, jméno **projektu** a **případu**. V levém dolním rohu je vypínání bublínkové nápovědy (*Tool Tips*). Okno *Variation Confirmation* dovoluje vytvořit novou **variaci vstupního souboru**. Jednotlivé variace mají stejné jméno jako je uvedeno pro jednotlivé **formuláře** v předchozí tabulce, liší se příponami. **Dialogy** přidělí nové **variaci** číselnou příponu a uloží o tom informaci do **archivu** EDIT.ARC. Při uložení **případu** se informace o **variacích** přepíše do **archivu** PŘÍPAD.ARC, kde jsou dostupné pro pozdější otevření **případu**.

Prohlížení **uživatelských výstupních souborů** se provádí příkazy z menu **Výsledky** a následnou specifikací souboru *Výběr uživatelského souboru*. Podmínkou ovšem je, že jsme typ souboru zadali před spuštěním simulace příkazem **Modifikovat| Uživatelské soubory** při použití jmenovací konvence. Jmenovací konvence spočívá v zadáním generického jména pro **uživatelské soubory**. Není-li zadáno toto jméno použije program jméno **případu**.

Prohlížení standardních výstupních souborů s hlášením se provádí příkazy z menu **Hlášení**:

- \* **Inicializace** ⇒ INIC.HLA
- \* **Události** ⇒ AKCE.HLA
- \* **Změny kroku** ⇒ KROK.HLA
- \* **Počáteční podmínky** ⇒ UST.INI
- \* **Inicializace motorů** ⇒ vypíše hlášení o výpočtu počátečních podmínek u motorů
- \* **Kontrola** ⇒ vypíše hlášení o kontrole překročení zadaných mezí síťových veličin
- \* **Diagnostika logik** ⇒ vypíše hlášení o působení automatik a logik
- \* **Chod sítě** ⇒ vypíše hlášení pomocného programu UST.


Prohlížení výstupních souborů s analýzou se provádí těmito příkazy z menu **Analýza**:

- |   |  |
|---|--|
| <ul style="list-style-type: none"> <li>* <b>Síť</b> →</li> <li>* <b>Primární regulace</b> ⇒ PRIM_REG.ANA</li> <li>* <b>Secondární regulace</b> ⇒ SEK_REG.ANA</li> </ul> | <ul style="list-style-type: none"> <li>• <b>Počáteční</b> ⇒ SIT_POC.ANA</li> <li>• <b>Koncová</b> ⇒ SIT_KON.ANA</li> <li>• <b>Rozdíl počátek-konec</b> ⇒ SIT_ROZ.ANA</li> <li>• <b>Překročení limitů</b> ⇒ SIT_ROZ.ANA</li> <li>• <b>Výkonová rovnováha</b> ⇒ SIT_BAL.ANA</li> </ul> |
|---|--|

přičemž na příkaz **Network** se objeví vnořené menu s nabídkou jednotlivých typů analýz stavu sítě. Jedná se o počáteční a koncový stav, dále rozdíl mezi těmito stavy a výpis překročení mezí. Před spuštěním simulace je nutno zadat požadavek na analýzu příkazem **Modifikovat| Analýza**.


Spuštění dynamické simulace (výpočtu) lze provést dvojím způsobem:

1) příkazem **Simulace** z menu **Spust'**

2) stisknutím tlačítka 

Spuštění výpočtu ustáleného stavu (chodu) lze provést dvojím způsobem:

1) příkazem **Chod sítě** z menu **Spust'**

2) kliknutím na ikonu 

Ukončení práce v uživatelském rozhraní se provede příkazem **Soubor| Uzávěřit**.

Připomene některá obecná pravidla pro provádění příkazů a povelů v prostředí WINDOWS:

- příkaz z menu se provede kliknutím nebo stiskem <ALT X>, kde X je podtržené písmeno názvu
- povel v padacím menu se provede kliknutím nebo klávesou s písmenem podtrženým v názvu
- jestliže je daný povel vybrán (modře probarven) provede se stiskem klávesy <ENTER>, zároveň stavový řádek v dolní části okna zobrazuje jednořádkovou nápovědu.

Kontextová nápověda je dostupná dvojím způsobem:

- 1) kliknutím na ikonu s šipkou a otazníkem (v pruhu nástrojů) se dosavadní kurzor ve tvaru šipky změní na kurzor s otazníkem a kliknutím na příkaz nebo ikonu se dostaneme do nápovědy
- 2) stiskem klávesy F1 (před vykonáním daného příkazu) se dostaneme do režimu nápovědy.

Pro správnou funkci **dialogů** je nutno:

- nastavit v prostředí WINDOWS jako desetinný oddělovač tečku (pomocí příkazu Nastavení, Ovládací panely, Místní nastavení, Číslo)
- aktuální verze katalogu SYMBOL.CAT, musí být pouze v globálním podadresáři GLOBAL.DAT
- všechny řádkové komentáře (uvedené za data na řádku) musí být odděleny od dat vykřičníkem
- délky textových řetězců jmen, používaných pro křížové odkazy v **databázích** musí být shodné jako odpovídající jména v **tabulkách** a **katalozích**.

Příkazem **Volby** se nastaví používaný externí textový editor .a jméno uživatele.


## 2. VÝPOČTY KRÁTKODOBÉ STABILITY

Krátkodobá dynamika řeší otázky přechodové stability synchronního stroje, tj. schopnost zůstat v synchronním provozu se sítí. K ohrožení přechodové stability dochází po poruchách typů zkratů, takže hlavní náplní tohoto oddílu bude simulace různých typů zkratů. Cílem výpočtů je:

- zjištění mezní doby trvání zkratu
- zjištění trajektorie impedance viděné ochranou při zkratu.

Oba typy výpočtů slouží pro kontrolu nastavení ochran, které zajišťují odpojení zkratu.

Nejprve otevřeme **projekt** tím, že ve stromu projektů klikneme na jeho jméno a pak

stiskneme tlačítko .

V projektu SHORT je uloženo několik **případů**.

Jako výchozí zvolíme případ UST\_STAV. Vybereme ho kliknutím a s příkazem **Případ | Otevřít** se jménem ust\_stav. Postup je obdobný jako při otevírání **projektů** a také můžeme vidět

doplňující informace o **případu**: jméno autora, datum vzniku a stiskneme tlačítko .

Po spuštění simulace příkazem **Spust' Simulace** nebo příslušným tlačítkem ukáže **grafika** časové průběhy ustáleného stavu - tj. přímky rovnoběžné s osou času. To svědčí o správné inicializaci dynamického modelu a můžeme tedy přistoupit k simulaci zkratu.

## 2.1. Zadání zkratu pomocí scénáře

Pro simulaci poruch v ES slouží v programu MODES tzv. **scénář**. Je to časový sled **zásahů** zadávaný ve vstupním souboru SCENAR.DAT příkazem **Modifikovat| Scénář**.

V textovém režimu se zadávání zásahů provádí pomocí klíčových slov a každý **zásah** vyžaduje určité parametry. Zadávání zásahů ulehčuje tzv. **makro**, které má tvar \$(Jm\_makra:Retez1=Object) a plní dvojí funkci:

- přesměruje čtení do tabulky maker obsažené ve vstupním souboru MAKRA.DAT
- nahrazuje výskyt řetězce v tělu makra Retez1 řetězcem Object.

Zadávaní zásahů pomocí scénáře nejlépe pochopíme pomocí příkladů.

- ☐ po otevření souboru SCENAR.DAT, vložíme (zkopírováním předdefinovaného makra, uloženého v komentářové části souboru SCENAR.DAT). Jednotlivá makra a scénáře jsou odděleny hvězdičkovým řádkem s krátkým komentářem na začátek řádek s **makrem** 3\_foul-b, takže obsah souboru bude vypadat takto:

```
$(3_foul-b:bran=VET1)
10000,'END'
```

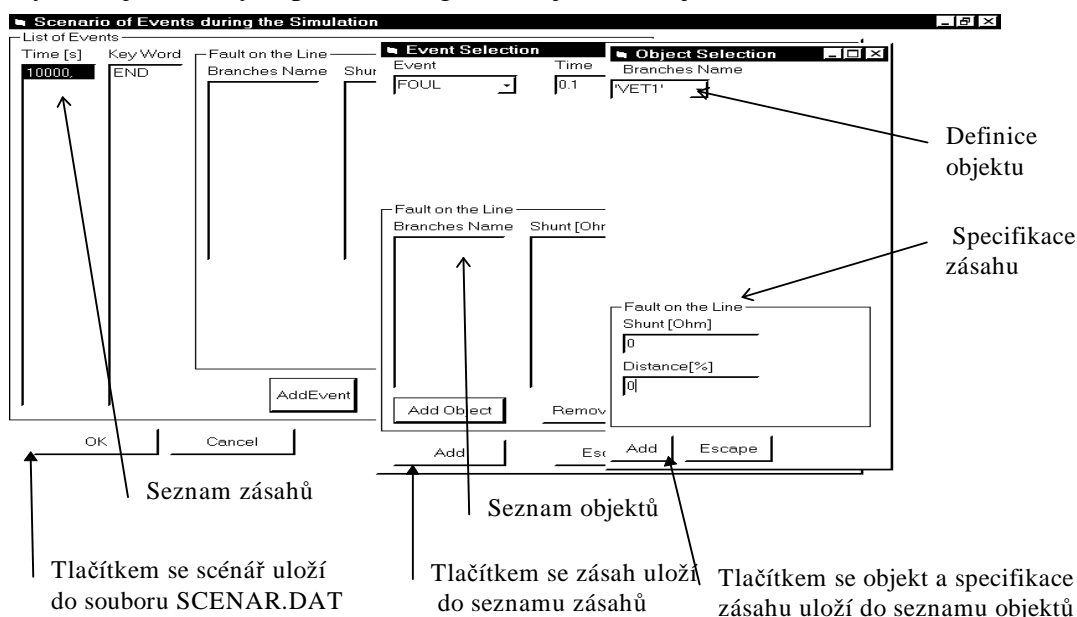
### Výp. 1 Výpis vstupního souboru SCENAR.DAT pro simulaci trojfázového zkratu

Jakmile program narazí na řádek začínající znakem \$ začne prohledávat tabulku maker MAKRA.DAT dokud nenalezne řádek s jménem **makra**. Pak načte následující řádky z tabulky maker (tělo makra) a nahradí v nich výskyt řetězce bran jménem větve VET1.

- ☑ V dialogovém režimu se dostaneme do dialogu *Scénář zásahů* a provedeme tyto operace:

- ☞ tlačítkem **AddEvent** se otevře okno *Event Selection*
- ☞ v rámečku *Object Type* zvolíme *Branch* a v seznamu vybereme *FOUL*
- ☞ v textovém poli *Time* zadáme čas zkratu 0.1
- ☞ tlačítkem **AddObject** se otevře okno *Object Selection* a vybereme **větev** VET1
- ☞ zadáme velikost šuntu 0 a vzdálenost zkratu od začátku větve také 0
- ☞ specifikaci zásahu potvrdíme tlačítkem **Add** a dialog opustíme tlačítkem **Escape**
- ☞ tlačítkem **Add** zařadíme zásah do **scénáře** a tlačítkem **Escape** ukončíme výběr **zásahů**
- ☞ po stisku tlačítka **OK** se objeví žádost o potvrzení vytvoření nové variace
- ☞ po stisku tlačítka **Yes** se **scénář** uloží do variace SCENAR.010.

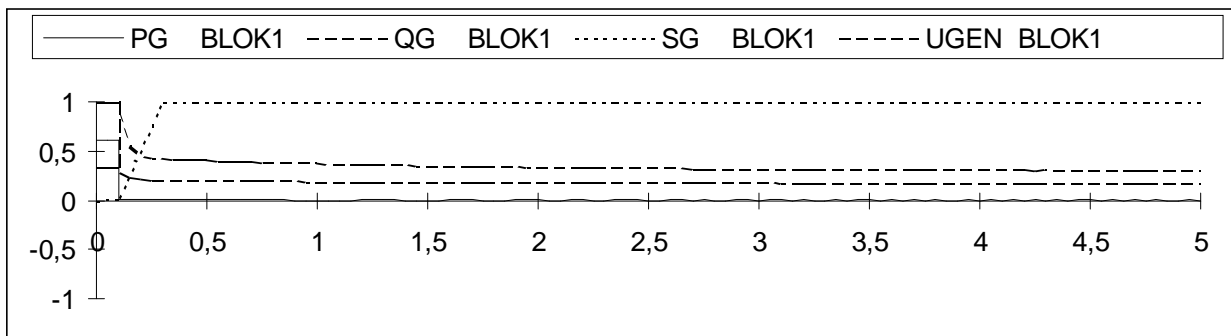
Význam jednotlivých prvků dialogu ukazuje následující obrázek:



### Dial. 1 Prvky dialogu zadávání scénáře a jejich význam

Uživatel také může zvolit pohodlnější způsob tím, že otevře předdefinovaný **případ** Zkrat postupem popsáním v úvodu oddílu.

Po opětovném spuštění programu nám grafika ukáže obrázek podobný tomuto:



**Obr. 1 Grafika se standardním výstupem při simulaci zkratu**

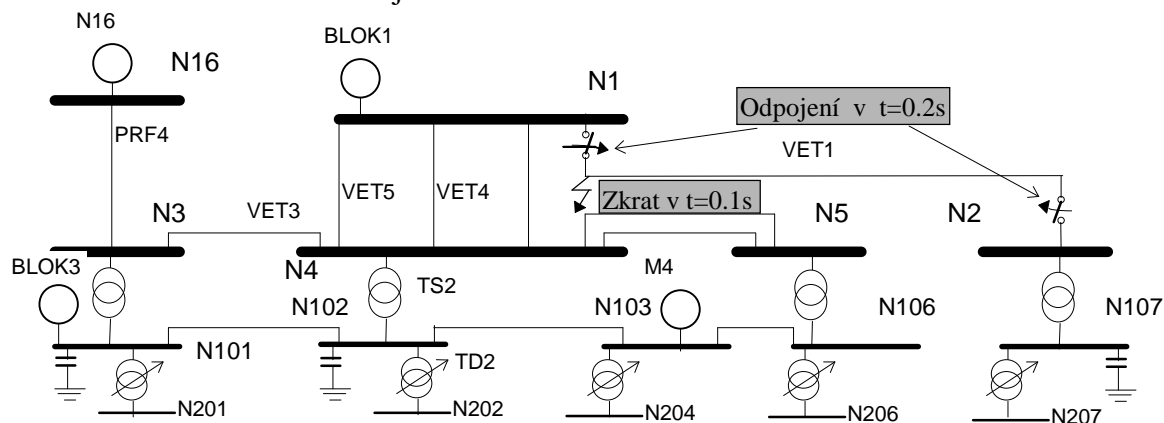
Na obrázku jsou vidět průběhy „standardních“ proměnných **bloku** BLOK1 při blízkém trojfázovém zkratu. Činný výkon klesá k nule, skluz generátoru naopak narůstá. Vyplývá to z fyzikální podstaty tohoto jevu, kdy poklesem činného výkonu (a tím i protimomentu) generátoru při nezměněném momentu turbíny (regulace turbíny nestačí rychle sledovat pokles činného výkonu) vzroste akcelerační moment a celé soustrojí je urychlováno (skluz roste). Pokud by porucha nebyla včas likvidována, dosáhly by otáčky nebezpečných hodnot a soustrojí by havarovalo.

Svorkové napětí pokleslo na 18% (tvoří ho úbytek na blokovém trafu). Pokles napětí brzdí pohony ve vlastní spotřebě, takže hrozí jejich zastavení.

Pokud by porucha trvala déle, rotor by se urychlil natolik, že po jejím odpojení by synchronizační síly nevtáhly stroj zpátky do synchronního chodu a stroj by přešel do asynchronního chodu. Tento svými rázy ohrožuje bezpečnost soustrojí, takže stroj musí být odpojen. Likvidace poruchy proto musí být provedena v nejkratším čase odpojením postižené **větve**.

## 2.2. Odstranění zkratu podle scénáře

V této kapitole se provede simulace třífázového zkratu a jeho odpojení. Porucha je zjednodušeně znázorněna na následujícím schématu:



**Sch. 1 Schéma zkratu a jeho odpojení**

Nejjednodušší způsob simulace odstranění zkratu je pomocí **scénáře**, kdy v předem definovaném čase zadáme zásah 'CLER' s parametrem  $I=0$ . Tento zásah odpojí postižené vedení ve všech fázích z obou stran.



¶ V textovém režimu je možné použít předdefinované makro s názvem `clear_foul`, takže obsah souboru `SCENAR.DAT` a odpovídající makra v tabulce `MAKRA.DAT` budou vypadat takto:

	0.1,'FOUL'
	1
	'bran',0,0
\$(3_foul-b:bran=VET1)	
\$(clear_foul:bran=VET1)	0.2,'CLER'
10000,'END'	1
!!!!!!!!!!!!!!!!!!!!!!!!!!!!	'bran',0

## Výp. 2 Výpis souboru `SCENAR.DAT` pro simulaci zkratu a jeho vypnutí

Uvedené dvě makra zadávají trojfázový kovový zkrat na začátku větve `VET1` a odepnutí větve `VET1` v čase 0.2s (tj. 0.1s po vzniku zkratu).

☑ V dialogovém režimu se zadání odepnutí zkratu provede obdobným způsobem, jak bylo popsáno v předchozí kapitole s tím, že se po výběru **zásahu** `CLER` zvolí vypnutí zkratu i větve `Fault+Line` a v textovém poli *Time* zadá čas odepnutí 0.2.

Odepnutí větve `VET1` bude signalizováno jednak probliknutím piktogramu **větve** a jednak hlášením v pravém horním rohu **grafiky**. Přesto si pro kontrolu vypíšeme např. průběh toku činného výkonu **větví** `VET1`. Proměnné, které se během výpočtu vypisují do **grafiku** upravíme příkazem **Modifikovat| Grafika**.

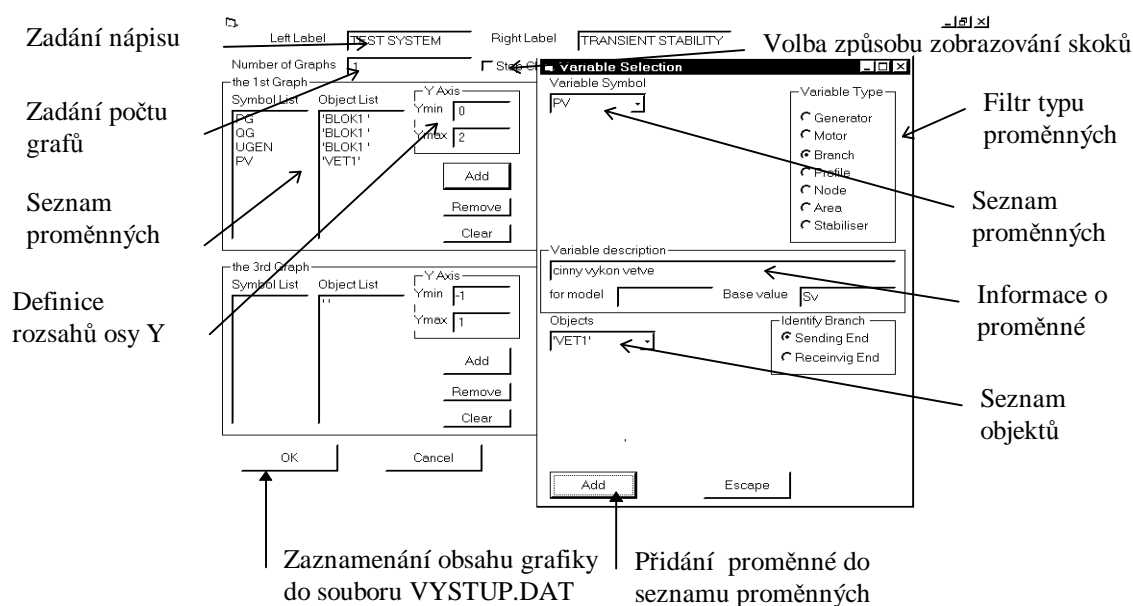
¶ V textovém režimu ve vstupním souboru `VYSTUP.DAT` vypíšeme předdefinovaná makra `FOULT_1` a `OBJ_UN+BR` a upravíme rozsah osy závisle proměnných v prvním grafu na 0 a 2. Úsek zadání 1.grafu ve vstupním souboru `VYSTUP.DAT` pak bude vypadat takto:

*****
Ymin,Ymax [pj] rozsahy os nezávisle promenných v 1.grafu {2xREAL}
0.0,2.
seznam symbolu velicin v 1.grafu {maximalne 7x4mistny retezec-viz vysvetleni A nize}
\$(FOULT_1:)
jmena prislusnych bloku/vetvi/uzlu/profilu {6x6/4/6/6mistnych retezcu}
\$(OBJ_UN+BR:unit1=BLOK1)

## Výp. 3 Výpis úseku vstupního souboru `VYSTUP.DAT` pro zadání proměnných v 1. grafu

- ☑ V dialogovém režimu se dostaneme do dialogu *Data grafiky* a dále:
  - ☞ kliknutím na tlačítko **Clear** zrušíme zadané proměnné
  - ☞ stisknutím tlačítka **Add Variable** se otevře okno *Variable Selection*
  - ☞ vybereme *Generator* z *Object Type*
  - ☞ vybereme symbol PG z kombo boxu *Variable Symbol*
  - ☞ objekt BLOK1 je již vybrán, takže stiskem tlačítka **Add** volbu potvrdíme
  - ☞ postup opakujeme pro proměnné QG a UGEN
  - ☞ pro tok činného výkonu PV větvi VET1 přepneme přepínač *Object Type* na *Branch*
  - ☞ vybereme symbol PV v seznamu *Variable Symbol* a větev VET1 v seznamu *Objects*
  - ☞ tlačítkem **Add** výběr proměnných potvrdíme a tlačítkem **Escape** se vrátíme do výchozího dialogu
  - ☞ v textovém poli *Ymin* a *Ymax* zadáme rozsah 0 a 2
  - ☞ tlačítkem **OK** se objeví žádost o potvrzení vytvoření nové variace
  - ☞ po stisku tlačítka **Yes** se **scénář** uloží do variace VYSTUP.007.

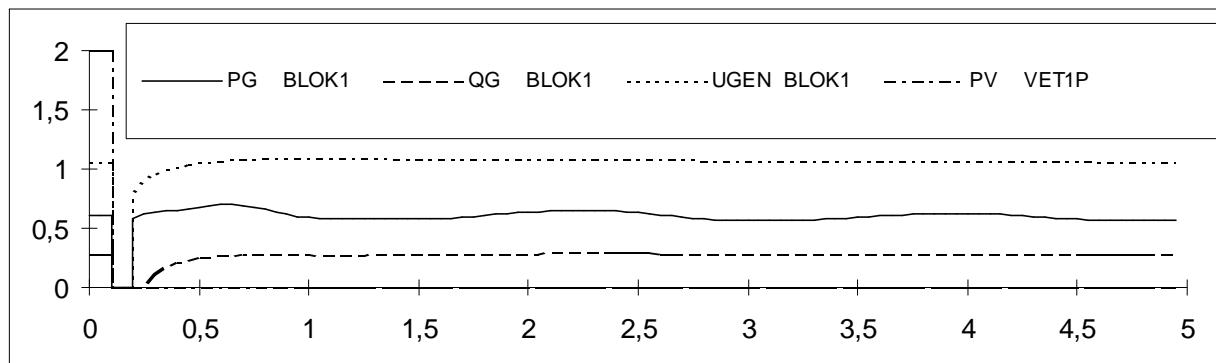
Význam jednotlivých prvků **dialogu** ukazuje následující obrázek:



## Dial. 2 Prvky dialogu zadávání obsahu a vzhledu grafiky a jejich význam

Uživatel také může zvolit pohodlnější způsob tím, že otevře předdefinovaný **případ** Zkrat\_3f, který obsahují informace o obou předchozích editacích. Obě **variace** musí být pochopitelně ve vstupním adresáři připraveny.

Po spuštění programu nám grafika ukáže obrázek podobný tomuto:



**Obr. 2 Průběhy proměnných bloku a toku činného výkonu větví při vypnutí zkratu**

Je vidět stabilní průběh po vypnutí zkratu. Doba vypnutí zkratu je dána scénářem, v našem případě byla 0.1 s. Tok činného výkonu větví VET1 během zkratu je nulový, jelikož jsme definovali třífázový kovový zkrat, takže napětí v místě zkratu kleslo na nulu. Po zkratu bude nulový rovněž, protože větev byla odpojena podle scénáře.

Postup jednotlivých zásahů se vypisuje do výstupního souboru AKCE.HLA, který je dostupný příkazem **Hlášení Události** a má po provedení výpočtu tento obsah:

Hlasení o udalostech:		
"CIGRE Belgium Test System"		
Cas [s]	Objekt	Udalost
0.100	VET1	zkrat na vetvi dle scenare
0.200	VET1	odpojena vetev dle scenare
0.200	VET1	vypadkem vznikl novy ostrov

**Hla. 1 Hlášení o provedení zásahů podle scénáře a vzniku ostrova (kontrola topologie)**

Soubor kromě hlavičky s komentářem (zadáva se ve vstupním souboru SOUBORY.DAT a slouží pro dokumentační účely) obsahuje časový sled akcí prováděných programem. Kromě očekávaných hlášení o zkratu a odpojení větve je zde informace o tom, že vznikl nový ostrov. Program při jakékoliv komutaci (změně stavu větve) v síti provádí kontrolu topologie a při vzniku izolované části soustavy - ostrova, vypíše hlášení do souboru AKCE.HLA. Podobně vypíše hlášení při zpětném přifázování ostrova k ostatní soustavě.

Dosud jsme ve scénáři zadali pevný čas, za který bude porucha odepnuta. Doba trvání poruchy byla zvolena ze zkušenosti tak, aby nedošlo ke ztrátě stability. Simulační program ovšem umožňuje určit mezní dobu trvání zkratu, při které ještě nedojde ke ztrátě stability. Tento údaj má velký praktický význam, neboť ochrany a automatiky musí zajistit odpojení poruchy do této doby. Program MODES používá k nalezení mezní doby trvání zkratu („Critical Clearing Time“) tzv. mechanismus návratu.

## 2.3. Nalezení mezní doby trvání zkratu pomocí návratu

Mechanismus návratu využívá zásah 'SNAP'. Tento zásah v zadaném čase uloží stav modelované soustavy do pomocného souboru a později ho může opět načíst - tedy vrátit se k předem definovanému stavu. To nám umožňuje provést řadu výpočtů na jedno spuštění programu.

▢ Příklad zadání tří cyklů návratu je v následujícím výpisu vstupního souboru SCENAR.DAT:

```
.1,'SNAP'
1
0
$(3_foul-b1:bran=VET1)
$(clear_fou1:bran=VET1)
5.,'SNAP'
1
1
$(3_foul-b2:bran=VET1)
$(clear_fou2:bran=VET1)
10.,'SNAP'
1
1
$(3_foul-b3:bran=VET1)
$(clear_fou3:bran=VET1)
10000,'END'
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

### Výp. 4 Zadání tří cyklů zkrat - vypnutí zkratu pomocí mechanismu návratu

Podle tohoto scénáře se zadá třikrát zkrat na větvi VET1 pomocí makra 3\_foul-b\* a jeho vypnutí pomocí makra clear\_fou\*, přičemž doba zkratu se postupně prodlužuje po 0.25s ze 0.25 na 0.75s (jak plyne z těla makra ve vstupním souboru MAKRA.DAT). Před zadáním makra 3\_foul-b\* se provede zápis stavu (v prvním cyklu) nebo návrat do něho (v dalších cyklech) zásahem 'SNAP'.

Jelikož požadovaná doba výpočtu se třikrát zvětší, je nutno upravit vstupní soubor RIZENI.DAT příkazem **Modifikovat|Řízení**.

▢ Po zadání Tkon=15s bude vstupní soubor RIZENI.DAT vypadat takto:

```
Soubor RIZENI.DAT pro zadání řídicích parametrů výpočtu
Ipoc: určuje způsob inicializace výpočtu
1
Tkon[s] - koncový čas výpočtu, Int - interval výpisu a kontroly
15, .05
```

### Výp. 5 Zadání prodlouženého času výpočtu do vstupního souboru RIZENI.DAT

Ve vstupním souboru RIZENI.DAT jsou uloženy řídicí parametry výpočtu. Kromě již zmíněného koncového času Tkon, se zde zadává i interval výpisu Int. V tomto intervalu se provádí výpis časových průběhů do grafiky a uživatelských výstupních souborů. Parametr Ipoc definuje režim inicializace výpočtu tj. způsob jak program zjistí počáteční stav modelované soustavy. Jsou možné tyto volby Ipoc:

- 1 - program načte napětíové poměry<sup>1</sup> a podle nich zatíží bloky<sup>2</sup>. Pak spočítá počáteční podmínky pro řešení soustavy algebraicko-diferenciálních rovnic a uloží je do souboru POCAT.SNP
- 0 - načítá počáteční podmínky přímo ze souboru POCAT.SNP
- 3 - spočítá počáteční podmínky, uloží je do souboru POCAT.SNP a ukončí výpočet
- -1 - načítá počáteční podmínky z koncového stavu předchozího výpočtu
- -3 - načítá počáteční podmínky ze stavu předchozího výpočtu uloženého zásahem 'SNAP'.

<sup>1</sup> ze vstupního souboru UST.DAT

<sup>2</sup> definované ve vstupním souboru GEN.DAT

V našem případě ponecháme režim s výpočtem počátečních podmínek při každém výpočtu.

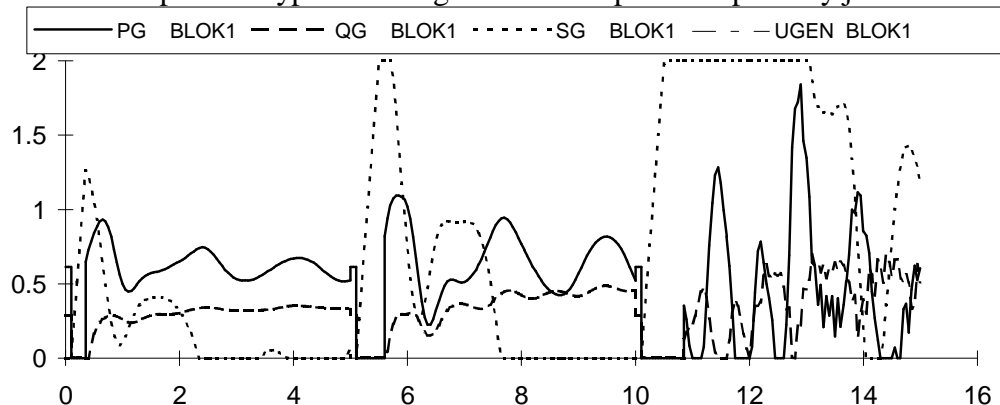
☒ V dialogovém režimu se dostaneme do dialogu *Data řízení výpočtu a*

☞ na kartě *Dynamic Calculation* napíšeme 15 do textového boxu *Time of Siumulation End*

☞ po stisku tlačítka **OK** se objeví žádost o potvrzení vytvoření nové variace

☞ po stisku tlačítka **Yes** se data uloží do variace RIZENI.003.

Uživatel může opět zvolit pohodlnější způsob tím, že otevře předdefinovaný **případ CCT\_15**. Po spuštění výpočtu nám grafika ukáže podobné průběhy jako na následujícím obrázku:



**Obr. 3 časový průběh proměnných při cyklickém výpočtu mezní doby trvání zkratu**

Pilovité průběhy proměnných při třetím cyklu ukazují, že k porušení stability došlo při době trvání zkratu 0.75s. Jelikož při druhém cyklu s dobou trvání 0.5s byly průběhy ještě stabilní, bude mezní doba ležet v intervalu 0.5 - 0.75s. Její upřesnění lze provést dalším spuštěním programu při upravených dobách trvání zkratu (např. 0.6, 0.65 a 0.7s).

Program MODES umožňuje při návratu použít i jiný způsob zobrazení v **grafice**, kdy se po **návratu** vrací i čas do zapamatované hodnoty. Pro tento způsob je nutno připravit **variaci** souboru RIZENI.DAT příkazem **Modifikovat| Řízení**.

☞ Na kartě *Dynamic Calculation* zkrátím čas výpočtu na 5.1 s

☞ zaškrtneme textový box *Keep the Time ..*

☞ opustíme dialog doporučeným způsobem s vytvořením **variace** RIZENI.004.

Jelikož se mění časy zadávání zásahů ve scénáři, upravíme variaci souboru KONT.DAT příkazem **Modifikovat| Scénář**. Všechny časy zadání poruchy budou tentokrát stejné 0.1 s. Časy vypnutí zkratu se budou při každém **návratu** postupně zvětšovat o 0.25s stejně jako v předchozím případě.

Ztrátu stability **bloku** jsme rozpoznali vizuálně pomocí typických „pilovitých“ časových průběhů na rozdíl od stabilních „kývavých“. Program umožňuje rozpoznat ztrátu stability jednotlivého stroje pomocí kontroly překročení mezního vnitřního zátěžného úhlu (mezi osou rotoru a fázorem svorkového napětí). Překročení úhlu 360 stupňů znamená jeden prokluz rotoru. Pro kontrolu stability na **blocích** připravíme **variaci** souboru KONT.DAT příkazem **Modifikovat| Kontrola**.

☒ V dialogovém režimu se dostaneme do dialogu *Data kontrolovaných proměnných*

☞ V pravé dolní části zadáme úhel 180 stupňů

☞ opustíme dialog doporučeným způsobem s vytvořením **variace** KONT.002.

Uživatel může opět zvolit pohodlnější způsob tím, že otevře předdefinovaný **případ CCT\_5**, který obsahuje informace o třech již upravených variacích.

Po puštění simulace jsou časové průběhy stejné jako v předchozím **případě**. Situace se změní po prvním **návratu**, kdy se časové průběhy vrací k nule (přesněji do času uloženého předchozím **zásahem** 'SNAP' s parametrem 0. Časové průběhy získávají vějířovitý charakter. Při třetím návratu lze pozorovat červené probliknutí piktogramu generátoru, který signalizuje ztrátu stability (přesněji prokluz generátoru). Hlášení o tom se zapisuje i do výstupního souboru do výstupního souboru KONT.HLA, který je dostupný příkazem **Hlášení| Kontrola**.

Při simulaci zkratů jsme zatím vycházeli ze základního modelu elektrizační soustavy. V dalším provedeme rozšíření modelu o tzv. **automatiky**.

## 2.4. Použití automatiky pro odpojení zkratu

Dosud jsme odstranění poruchy prováděli v předem zadaném čase zjednodušeně pomocí scénáře, aniž by nás zajímalo jak je provedeno ve skutečnosti. Další popis zaměříme na zpřesnění modelu, které nám umožňují tzv. **automatiky**.

V reálném systému odpojení zkratu provádí distanční ochrana vedení v součinnosti s příslušnými vypínači. Program MODES umožňuje modelovat tuto ochranu pomocí **automatiky**. **Automatika** měří vybranou **proměnnou** a při splnění určité podmínky provede **zásah**<sup>1</sup>. **Automatiky** se zadávají ve vstupním souboru AUTOMAT.DAT. Soubor je z uživatelského prostředí dostupný příkazem **Modifikovat|Automatiky**.

V souboru AUTOMAT.DAT jsou seznam automatik a jejich působení

Naut- počet automatik:

1

Poc. stav	Jmeno autom. [s]	Per. Symbol	Merena hodn. Misto	X Merena hodn. Y Misto	Klic Oper.	XNast/X0	Y0	R/K	N Tkon [s]	Tzas [s]	Tblok [s]	Pusobeni Kod	Objekt	Par Nautx	
0	'dmh_1c'	0.01	'RV'	'VET1'	'XV'	'VET1'	'CIRC'	'LT'	28.	28.	40.0	0.025	.1	10	'BRAN' 'VET1' 0

### Výp. 6 Výpis vst.souboru AUTOMAT.DAT se zadáním distanční ochrany

**Automatika** dmh\_1c měří polohu impedance viděné ochranou v kartézské rovině. Pokud se koncový bod octne v zakázané oblasti (definované vysunutou kružnicí<sup>2</sup>) po dobu Tkon=0.025s, dojde se zpožděním Tzas=0.1 k zásahu 'BRAN' - vypnutí větve VET1.

■ Ve vstupním souboru RIZENI.DAT je upraven koncový čas výpočtu na 5s:

Soubor RIZENI.DAT pro zadání řídicích parametrů výpočtu

Ipoc: určuje způsob inicializace výpočtu

1

Tkon[s] - koncový čas výpočtu, Int - interval vypisu a kontroly

5, .05

Dx, Dxmin, Dxmax[s] - počáteční, minimální a maximální integrační krok

0.05, .01, .1

pro MODES: eps[pu], epsU, ltermmax, Diagn

.002, .002, 10, 0

### Výp. 7 Úsek vstupního souboru RIZENI.DAT se zadáním parametrů výpočtu

Počáteční krok dx by měl být násobkem minimálního kroku dxmin, jinak program automaticky minimální krok upraví a vypíše o tom hlášení do souboru INIC.HLA. pV našem případě tedy bude minimální hodnota kroku změněna na 0.0125. **Automatika** je ve výchozím stavu blokována (poc.stav=0). Aby byl zkrat odpojen automatikou je nutno, ji odblokovat zásahem AUT s parametrem 1 v čase t=0, což je provedeno ve variaci scénáře SCENAR.005. Všechny data jsou připraveny **případu ZkratAut**.

Po otevření případu a spuštění programu nám **grafika** ukáže obdobný obrázek jako v kapitole 2.2 Ke změně došlo v souboru AKCE.HLA, který obsahuje následující hlášení:

Hlášení o událostech:

"CIGRE Belgium Test System "

Cas [s] Objekt Událost

0.100 VET1 zkrat na vetvi dle scenare

0.250 VET1 vypnuto automatikou:dmh\_1c

0.250 VET1 vypadkem vznikl novy ostrov

### Hla. 2 Hlášení o provedení zkratu podle scénáře a jeho vypnutí činností automatiky

Podle očekávání byla větev vypnuta **automatikou** dmh\_1c, zásah je však proveden až za 0.15s po vzniku poruchy a ne za očekávaných 0.125 s (Tkon+Tzas). Je to způsobeno tím, že v programu se čas nemodeluje spojitě, ale v diskretních krocích. Následující kapitola pojedná o tomto problému podrobněji.

<sup>1</sup> funkce automatik je popsána v Popisu [1] v kapitole Model automatik.

<sup>2</sup> charakteristika „MHO“

## 2.5. Modelování času

V programu MODES se čas nemění spojitě, ale po těchto krocích:

- Integrační krok Dx pro řešení diferenciálních rovnic - definuje se v souboru RIZENI.DAT
- Interval výpisu Int pro výpis **proměnných** (do **grafiky a uživatelských výstupních souborů**) a kontrolu síťových veličin<sup>1</sup> - definuje se také ve souboru RIZENI.DAT
- Perioda **automatik/logik** Per - definuje se v souborech AUTOMAT/LOGIKY.DAT
- Vzorkovací krok Tsamp pro centrální regulátor P/f - definuje se v souboru AUTSEK.DAT.

Změnu integračního kroku Dx provádí program automaticky tak, aby dodržel zadanou lokální chybu integrační metody<sup>2</sup> eps (zadáva se spolu s minimálním a maximálním krokem Dxmin a Dxmax ve vstupním souboru RIZENI.DAT). Hlášení o změnách program zapisuje do výstupního souboru KROK.HLA. Interval výpisu Int může být měněn zásahem 'SAMP' a integrační krok zásahem 'STEP'. Zbylé kroky Per a Tsamp nemohou být měněny během výpočtu. Mezi jednotlivými kroky platí tyto vztahy:

- 1) je-li interval výpisu Int menší než integrační krok Dx, má Dx prioritu ( $\text{Int} \Leftarrow \text{Dx}$ )
- 2) je-li perioda automatik Per menší než aktuální integrační krok Dx, má Dx prioritu ( $\text{Per} \Leftarrow \text{Dx}$ )
- 3) je-li vzorkovací krok Tsamp menší než aktuální interval výpisu Int, má Int prioritu ( $\text{Tsamp} \Leftarrow \text{Int}$ )

Jelikož program počáteční zadaný integrační krok (Viz **Chyba! Nenalezen zdroj odkazů.**) nemění (výstupní soubor KROK.HLA neobsahuje hlášení o jeho změně), platí pravidlo 2) neboť  $\text{Dx}=0.05 > \text{Per}=0.01$ .

Tím je tedy vysvětleno to, že k vypnutí nedošlo v čase 0.24s, ale až v čase 0.25s jako nejbližším násobku integračního kroku Dx. Pro zpřesnění modelu použijeme ve **scénáři zásah** 'STEP', který zajistí zmenšení integračního kroku na 0.0125 a tím správnou funkci **automatiky**.

☐ V textovém režimu bude obsah variace souboru SCENAR.013 vypadat takto:

```
0,'AUTO'
1
'dmh_1c' , 1 , 0
.1,'STEP'
1
0.0125 , 0
.1,'FOUL'
1
'VET1' , 0 , 0
10000,'END'
!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

### Výp. 8 Výpis vstupního souboru SCENAR.DAT se zadáním změny intervalu výpisu

Z výpisu je vidět, že interval Int se mění jednou před zadáním makra s poruchou a podruhé po vypnutí zkratu v čase 0.3. Druhý parametr zásahu 'SAMP' má význam zadané lokální chyby integrační metody. V době poruchy byla lokální chyba zmenšena na polovinu.

☒ v dialogovém režimu je možno zásah STEP přidat následujícím postupem v dialogu *Scénář zásahů*

☞ tlačítkem **Přidat zásah** se dostaneme do dialogu *Event Selection*

☞ v rámečku *Typ objektu* zvolíme *Čas* a v seznamu vybereme **zásah STEP**

☞ v textových polích zadáme maximální integrační krok *dxmax* 0.0125

☞ specifikaci zásahu potvrdíme tlačítkem **Přidat** a dialog opustíme tlačítkem **Zruš**

☞ tlačítkem **Přidat** zařadíme **zásah** do **scénáře** a tlačítkem **OK** kdialog opustíme

|| Můžeme se použít předdefinovaný **případ** AUT\_STEP.

Po spuštění programu bude automatika fungovat podle našich představ, jak se přesvědčíme v souboru AKCE.HLA.

<sup>1</sup> zadávanou ve vst.souboru KONT.DAT viz Uživatelské příručka, kap. Soubory popisující funkce programu

<sup>2</sup> podrobnosti o změně int.kroku jsou v Uživatelské příručce v kap. Soubory popisující funkce programu

## 2.6. Zobrazení v kartézské souřadné rovině (KR)

Dosud byly grafy na obrazovce ve formě časových průběhů **proměnných**. Program však umožňuje zobrazit fázovou **trajektorii** koncových bodů fázorů vybraných **proměnných**. Zadání se provede ve vstupního souboru VYSTUP.DAT po příkazu **Modifikovat| Grafika**.

☐ Pro zobrazení v kartézské rovině musíme ve vstupního souboru VYSTUP.DAT definovat:

- ☞ kartézský typ grafu - pomocí parametru  $N_{gr}=5$
- ☞ rozsahy reálné a imaginární osy- zvolíme hodnoty -20 a  $100\Omega$  tj. -0.0138 a 0.0693 pj.
- ☞ symbol proměnné 'Z' pro impedanci viděnou ochranou a objekty VET1 a VET1K (konec větve).

Upravený úsek vstupního souboru VYSTUP.DAT bude vypadat takto:

```
N_gr: Pocet grafu, Zmeny:0 bez skok.zmen,1 se skok.zmenami
5,1
.....
Ymin,Ymax [pj] rozsahy os zavisle promennych v 1.grafu/realne osy {2xREAL}
-0.0138,0.0693
seznam symbolu velicin v 1.grafu {maximalne 7x4mistny retezec-viz vysvetleni A nize}
'Z'
jmena prislusnych bloku/vetvi/uzlu/profilu {6x6/4/6/6mistnych retezcu}
'VET1'
Ymin,Ymax [pj] rozsahy os zavisle promennych v 2.grafu/imaginarni osy {2xREAL}
-0.0138,0.0693
seznam symbolu velicin v 2.grafu
'Z'
jmena prislusnych bloku/vetvi/uzlu/profilu {6/4/6/6mistne retezce}
'VET1K'
```

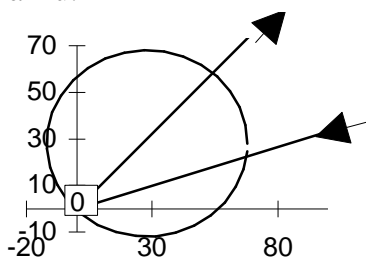
### Výp. 9 Úsek vstupního souboru VYSTUP.DAT upravený pro kartézský typ grafu

- ☑ V dialogovém režimu se dostaneme do **dialogu Data grafiky**, kde zadáme i kružnici, která tvoří hranici dovolené části **KR automatiky**  $dmh_{1c}$ :
- ☞ zaškrtneme box *Complex Plane*
- ☞ do textových boxů  $Xmin, Xmax$  a  $Ymin, Ymax$  napíšeme obdobné rozsahy jako u textového režimu
- ☞ v rámci *1st Graph* stiskneme **Clear** a tlačítkem **AddVariables** se dostaneme do *Variable Selection*
- ☞ vybereme *Branch* z *Object Type*, **proměnnou Z** a **objekt VET1**, stiskneme tlačítka **Add** a **Escape**
- ☞ obdobně postupujeme pro *2nd Graph* s rozdílem, že zvolíme *Receiving end*
- ☞ v rámci *3rd Graph* stiskneme **Clear** a **AddVariables**, dostaneme do *Selection of Static Object*
- ☞ zadáme souřadnice bodů -0.00831, -0.00831 a 0.0471, 0.0471 a vybereme CIRCLE
- ☞ stiskneme tlačítka **Add** a **Escap** a dialog ukončíme.

Program umožňuje udělat značku na **trajektorii** v KR. Provede se to **zásahem** 'MARK' se zadaným objektem '1' (pro první trajektorii v grafice) a parametrem 0 (značka 0) ve **scénáři**.

Změny jsou provedeny ve **variacích** VYSTUP.002 a SCENAR.006, takže postačí otevřít **případ Komplex**.

Po spuštění programu **grafika** ukáže průběh fázové **trajektorie** obdobný následujícímu obrázku:



Rozdíl v obrázku je v cejchování os v  $\Omega$  (grafika je poměrných jednotkách). Po vniknutí do vnitřku kružnice dojde k odepnutí větve a fázor se přesune do „nekonečna“. Svědčí o tom i přetečení čísel okamžitých hodnot. Pozornému pozorovateli neujde, že trajektorie je jen jedna, zatímco jsme zadávali dvě. Druhá trajektorie je mimo definovaný rozsah os. Pokud rozsahy zvětšíme na  $\pm 0.693$  bude viditelná i druhá trajektorie.

**Obr. 4** Trajektorie zdánlivé impedance na začátku VET1 s charakteristikou automatiky



## 2.7. Simulace různých typů zkratů

Dosavadní výpočty simulovaly dokonalý (kovový) trojfázový zkrat na začátku větve (rovná se prakticky přípojnícovému zkratu). Zásah 'FOUL' má dva parametry:

- Vzdal - vzdálenost (v %) místa zkratu od začátku vedení v rozsahu 0 (začátek) -100 (konec)
- Xboc - velikost náhradní reaktance<sup>1</sup> (v  $\Omega$ ) připínané mezi místo zkratu a zem.

Zatímco první parametr umožňuje simulovat zkrat v kterémkoli místě větve, druhý parametr umožňuje simulovat jednak nesymetrické zkraty a jednak obloukový zkrat (zadáme-li zápornou hodnotu, je tato vyhodnocena programem jako činný odpor).

Simulace nesymetrických poruch (jedno a dvoupolové zkraty) vyžaduje bližší vysvětlení. Program MODES provádí výpočty pouze v sousledné složkové soustavě. Nesymetrické zkraty, jak známo z teorie složkových soustav, se projeví v sousledné složkové soustavě právě určitou reaktancí mezi místem zkratu a zemí (šuntem). Uvedený mechanismus lze použít na výpočty tzv. přechodové stability synchronních strojů. Zde se zjednodušeně předpokládá, že ostatní složky (nulová a zpětná) do statorového vinutí nepronikne nebo se její účinek zanedbává. Mechanismus však nelze použít pro přesné nastavení ochran měřících proudy nebo zdánlivé impedance (nadproudové nebo distanční ochrany), protože v programu se uvažuje pouze sousledná složka těchto proudů a impedancí. Proto příklad simulace nesymetrického zkratu je nutno brát pouze jako ilustrativní.

■ Ve vstupním souboru SCENAR.DAT jsou připraveny další tři předdefinovaná makra:

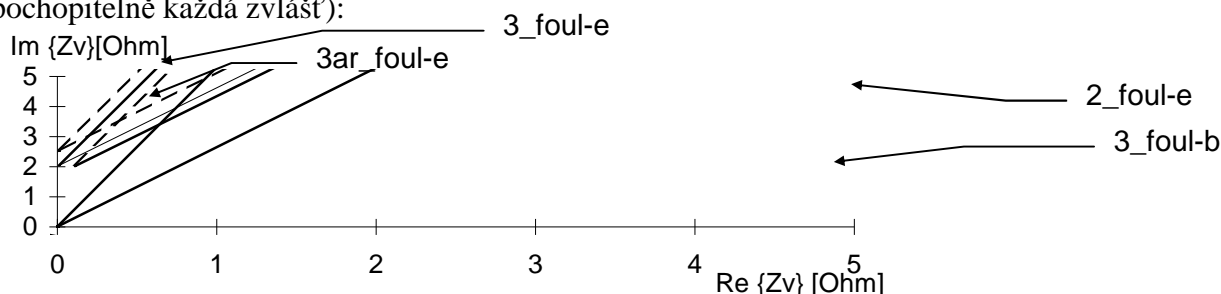
- ♦ 3\_foul-e .....trojfázový kovový zkrat na konci větve
- ♦ 3ar\_foul-e .....trojfázový obloukový zkrat na konci větve
- ♦ 2\_foul-e .....nesymetrický zkrat na konci větve simulovaný pomocí šuntu.

Čtenář si může postupně všechny poruchy vyzkoušet tím, že původní název **makra 3\_foul-e** přepíše výše uvedenými jmény (zbytek definice makra zůstává stejný) nebo řádek s původní definicí makra vymaže a zkopíruje místo něho připravená makra z komentářové části souboru.

☑ V dialogovém režimu je možno otevřít **případy** se stejnými jmény jako jsou výše uvedená **makra**. V druhém **případě** s obloukovým zkratem je jako statický objekt kreslena přímka rovnoběžná s osou imaginární (odporová charakteristika  $R=0.1 \Omega$ ). V třetím **případě** s nesymetrickým zkratem je jako statický objekt kreslena přímka rovnoběžná s osou reálnou (reaktanční charakteristika  $X=0.5 \Omega$ ).

Všechny uvedené poruchy jsou řádně vypnuty automatikou dmh\_1c, takže teoreticky by pro ochranu větve VET1 postačila jedna distanční ochrana umístěná na začátku větve. Situace je totiž zjednodušena tím, že větev tvoří paprskové vedení napájené z jedné strany, jak vyplývá ze schématu **Chyba! Nenalezen zdroj odkazů..** Ke složitějším případům zauzlené sítě se vrátíme později.

Na následujícím obrázku je detail trajektorií pro jednotlivé poruchy (při výpočtu jsou vidět pochopitelně každá zvlášť):



**Obr. 5 Trajektorie zdánlivé impedance pro různé typy zkratů (detail prvního kvadrantu)**

Čtenáře jistě napadlo, jestli lze časové průběhy zachovat pro pozdější zpracování a archivaci. Je to možné pomocí tzv. uživatelských výstupních souborů. O tom pojednává další kapitola.

<sup>1</sup> jestliže je zadaná hodnota záporná bude se jednat o činný odpor

## 2.8. Archivace časových průběhů proměnných

Dosud jsme se při výkladu setkali s **vstupními soubory** (obsahují vstupní data modelované soustavy a definují funkce programu) a se standardními výstupními soubory `AKCE.HLA`<sup>1</sup> a `KROK.HLA`<sup>2</sup>, které program zakládá automaticky ve výstupním adresáři `VYST`. Další tzv. **uživatelské výstupní soubory** program vytváří na žádost uživatele. Požadavek na trvalé ukládání vybraných časových průběhů se zadává před spuštěním programu ve vstupním souboru `SOUBORY.DAT`. Lze si vybrat z těchto typů (jsou uvedeny i kódy v apostrofech):

- 'ELE' základní elektrické veličiny bloku
- 'MEC' mechanické veličiny bloku
- 'MOT' veličiny motoru
- 'P','Q','I' toky a proudy větví, uzlů a profilů
- 'RX' reálné a imaginární složky zdánlivé impedance (viděné ochranou) vedení
- 'ABS' moduly napětí v uzlech definovaných v `KONT.DAT`
- 'ARG' argumenty napětí v uzlech definovaných v `KONT.DAT`
- 'ANG' absolutní nebo vzájemné úhly rotoru generátorů definovaných v `KONT.DAT`
- 'DF' odchylky frekvence napětí v aktivních uzlech definovaných v `KONT.DAT`
- 'SET' sada vybraných proměnných bloku
- 'LOS' ztráty ve větvích
- 'D12' stejné proměnné jako v 1. a 2. grafu na displeji
- 'D34' stejné proměnné jako v 3. a 4. grafu na displeji
- 'TAP' odbočky regulačních traf
- 'RAT' poměrné převody větví
- 'LFC' proměnné centrálního regulátoru P a f
- GR1-GR4 proměnné z 1. - 4. grafu

U prvních tří typů musíme specifikovat jméno bloku. U typu 'LFC' se specifikuje jméno oblasti. U ostatních typů program vybere příslušné objekty sám podle následujících pravidel:

- 1) vybrané proměnné bloku definované ve vstupním souboru `SOUBORY.DAT` u typu 'SET'
- 2) regulační trafa definované ve vstupním souboru `VET.DAT` u typu 'TAP'
- 3) pro ostatní typy uzly, větve a profily:
  - definované jako kontrolované ve vstupním souboru `KONT.DAT`
  - definované pro výstup do grafiky ve vstupním souboru `VYSTUP.DAT`
  - definované jako vstupy pro automatiky ve vstupním souboru `AUTOMAT.DAT`
  - profily definované ve vstupním souboru `UST.DAT`.

Program zjednodušuje zadání uživatelských výstupních souborů pomocí tzv. jmenovací konvence. Tato konvence je založena na těchto pravidlech:

- soubory se uloží do standardního podadresáře `VYST`
- u souborů typu 'ELE', 'MEC', 'MOT' a 'LFC' je jméno souboru stejné jako zadané jméno objektu
- u ostatních typů je jméno souboru shodné s generickým jménem (definuje se v `SOUBORY.DAT`)
- není-li zadáno generické jméno, použije se pro jméno souboru použije jméno **případu**
- přípona souboru je shodná s kódem typu souboru.

Tuto konvenci lze přepsat tím, že za kód (případně jméno objektu) zadáme v apostrofech kompletní specifikaci souboru tj.: 'Disk:\Cesta\Jméno.Přípona'.

<sup>1</sup> s hlášením o zásazích a akcích automatické regulace odboček traf a frekvenčního odlehčování zátěže

<sup>2</sup> s hlášením o automatické změně integračního kroku.

Úprava vstupního souboru SOUBORY.DAT se provede příkazem **Modifikovat | Uživatelské soubory**.

■ Následující úsek vstupního souboru definuje dva uživatelské výstupní soubory:

```
*****
Komentarovy radek do vyst.souboru {41mistne retezce} a jmeno pripadu {8mistny retezec}
                                                                 2x
'CIGRE Belgium Test System'
'OBR'
.....
zadani typu vystupnich souboru viz pozn. B nize
Nsoub- pocet vystupnich souboru (MAX.10)
2
Kod typu{3mistny retez.} [jmeno objektu (pouze pro MEC ELE MOT LFC {6mistny retez.})
(viz vysvetleni B )
_____Nsoubx-pro Nsoub>0
'D12'
'RX'
.....
```

### Výp. 10 Úsek vst.souboru SOUBORY.DAT s definicí uživatelských výstupních souborů

☑ V dialogovém režimu se otevře **dialog** zadání **Uživatelských výstupních souborů**:

☞ stisknutím tlačítko **Add Use File** se dostaneme do okna *Type of User File*

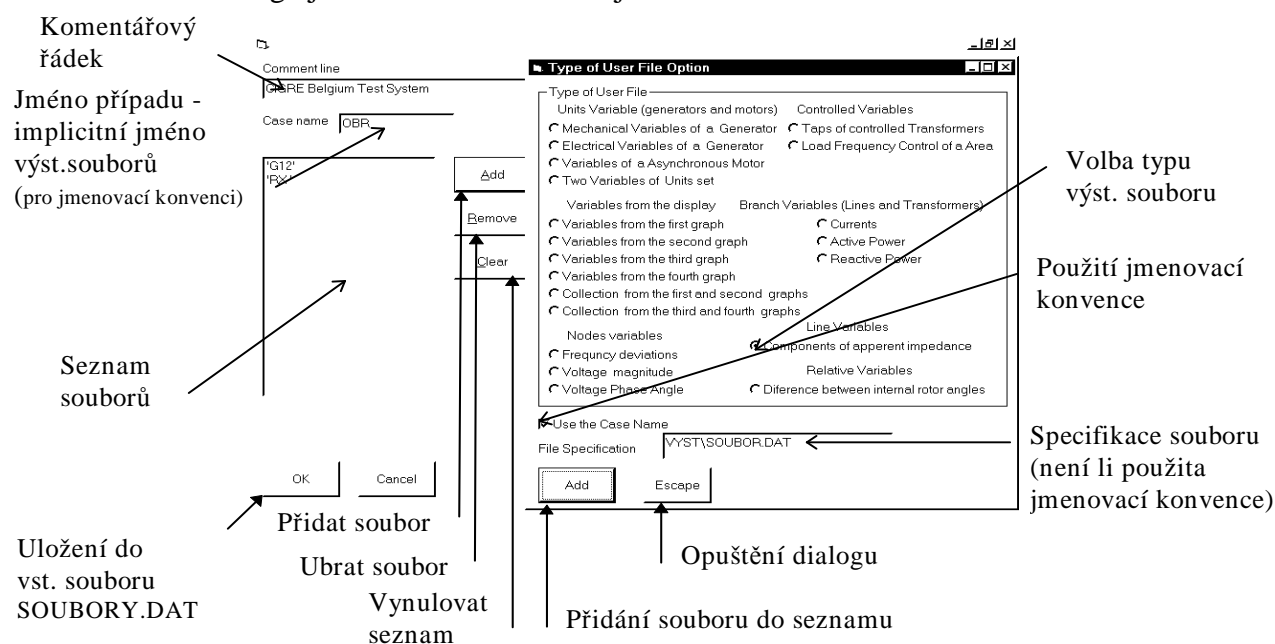
☞ na kartě *Variables from Display* se zvolí *Collection from the first..* a potvrdí tlačítkem **Add**

☞ postup opakujeme na kartě *Větve* volbou *Components of Apparent..* a stiskneme **Escape**

☞ V textovém poli *Generic Name* zadáme jméno OBR

☞ Dialog ukončíme stiskem tlačítka **OK**.

Vzhled dialogu je znázorněn na následujícím obrázku:



### Dial. 3 Dialog pro výběr uživatelských výstupních souborů a význam jeho prvků

Pro lepší názornost změníme ve **vstupním souboru** VYSTUP.DAT zobrazení v komplexní rovině na zobrazení jednoho grafu s **proměnnými** 'PG', 'QG', 'UGEN' pro **objekt** 'BLOK1'.

Změny jsou provedeny ve **variacích** VYSTUP.005 a SOUBORY.001, takže stačí otevřít **případ** s\_gener.

Jelikož je použito implicitní pojmenování pomocí generického jména OBR vzniknou v podadresáři VYST dva soubory se jmény OBR.D12 a OBR.RX. **Uživatelské výstupní soubory** jsou dostupné příkazem **Výsledky**. V padacím menu si vybereme příslušný typ (v našem případě např. *Display 1 & 2*) a dostaneme se do okna *Otevřít Result File*. V kombo boxu vybereme příslušný soubor (OBR.D12) a volbu potvrdíme tlačítkem **OK**.

Obsah souboru OBR .D12 je následující:

```
'CIGRE Belgium Test System - Transient Stab'
'Vypis promennych [pj] 1.grafu
'Yref', 1100.00, 1100.00, 24000.00,
' ', 'MW', 'MVar', 'V',
'Obj.', 'BLOK1', 'BLOK1', 'BLOK1',
't[s]', 'PG', 'QG', 'UGEN',
0.000, 0.61379, 0.33170, 1.08176,
.
0.100, 0.61381, 0.33169, 1.08176,
0.100, 0.04385, 1.09355, 0.43994,
.
0.240, 0.07482, 0.60003, 0.33111,
0.240, 0.59322, 0.02815, 0.84148,
```

### Výp. 11 Výpis uživatelského výstupního souboru typu D12

První řádek odpovídá komentáři zadanému v souboru SOUBORY.DAT. Druhý řádek specifikuje typ **výstupního uživatelského souboru**.

Další čtyři řádky definují jednotlivé **proměnné**. Jelikož jsou proměnné v poměrných hodnotách obsahuje třetí a čtvrtý řádek seznam vztažných hodnot (Yref) a jednotky. Pátý a šestý řádek obsahuje seznam **objektů** a symbolů **proměnných**.

Dále soubor obsahuje první sloupec s časem (nezávisle proměnná) a další sloupce s časovými průběhy (závisle proměnné). Sloupce jsou odděleny čárkami.

Jmenovací konvence umožňuje i druhy způsob implicitního pojmenování **uživatelských výstupních souborů** pomocí jména **případu**. Jestliže nezadáme generické jméno ve **vstupním souboru** SOUBORY.DAT. V textovém režimu se to provede zadáním prázdného řetězce pro generické jméno (' '). V dialogovém režimu se nechá textové pole *Generic Name* prázdné. Je možno zadat i výpisy **proměnných** z jednotlivých grafů pomocí typů GR1 až GR4. V dialogovém režimu se na kartě na kartě *Variables from Display* se zvolí *Variables from the first Graph*.

Změny jsou provedeny ve **variaci** SOUBORY.002, takže postačí otevřít **případ** s\_Prippj.

Obsah souboru S\_PRIPPJ.GR1 je následující:

```
t[s],PG _BLOK1 [p.j.],QG _BLOK1 [p.j.],UGEN_BLOK1 [p.j.],
0.000, 0.61379, 0.33170, 1.08176,
```

### Výp. 12 Výpis uživatelského výstupního souboru typu GR1

Hlavička souboru je zmenšena na jeden řádek, obsahující informace o proměnných. V souborech typu GR1 až GR4 je možno zadat alternativně výpis v pojmenovaných hodnotách.

Provede se to ve **vstupním souboru** VYSTUP.DAT V textovém režimu se to provede napsáním hvězdičky v symbolu proměnné (např. '\*PG'). V dialogovém režimu se zaškrtně *use Named Value* v dialogu *Variable Selection*.

Za povšimnutí stojí, že časy, kdy došlo ke komutaci (zkrat a odpojení **větvě**), se zapisují dvakrát (v t. před provedením **zásahu** a t<sub>+</sub> po provedení). Dvojí zápis se dá odstranit ve **vstupním souboru** VYSTUP.DAT. V textovém režimu se to provede zadáním parametru Zmeny=0. V dialogovém režimu se zruší proškrtnutí *Step Changes*.

Změny jsou provedeny ve **variaci** VYSTUP.006, takže postačí otevřít **případ** s\_Prip0.

Obsah souboru S\_PRIP0.GR1 je následující:

```
t[s], UGEN_BLOK1 [p.j.],UGEN_BLOK1 [V],QG _BLOK1 [MVar],QG _BLOK1 [p.j.],.....
0.000, 1.08176, 25962.25000, 364.87463, 0.33170, 0.61379, 675.16620,
0.100, 1.08176, 25962.13477, 364.86246, 0.33169, 0.61381, 675.19580,
.
0.240, 0.33111, 7946.60303, 660.02759, 0.60003, 0.07482, 82.29907,
```

### Výp. 13 Výpis uživatelského výstupního souboru typu GR1

Je vidět, že do souboru jsou vypisovány veličiny v poměrných i pojmenovaných hodnotách.

Dosud jsme testovali jedinou automatiku simulující distanční ochranu s charakteristikou „MHO“ - kružnicí vysunutou do 1.kvadrantu. S dalšími možnostmi modelování ochrany se seznámíme v 2.díle.

Tímto jsme vyčerpali základní témata simulace dynamické stability a následuje stručné shrnutí uvedené problematiky.

✍ Důležité body:

- Simulace poruch se provádí pomocí **zásahů** ve **scénáři**.
- **Scénář** se definuje v sekvenčním vstupním souboru SCENAR . DAT.
- Zadáání **zásahu** zjednodušují předdefinovaná **makra**, které mají tvar \$(Jm\_makra:Retez1=Object)
- Předdefinovaná **makra** jsou umístěna v tabulce **maker** MAKRA . DAT.
- Hlášení o **událostech** během výpočtu jsou uložena ve výstupním souboru AKCE . HLA.
- Program provádí kontrolu topologie sítě a hlásí vznik **ostrovů** v souboru AKCE . HLA.
- Mechanismus **návratu** provádí pomocí zásahu 'SNAP' cyklické výpočty na jedno spuštění.
- **Automatiky** slouží pro simulaci ochranných a řídicích prvků.
- Předdefinované **automatiky** se aktivují pomocí parametru Naut v hlavičce tabulky **automatik**.
- Tabulka **automatik** je umístěna ve vstupním souboru AUTOMAT . DAT .
- Měřicí články **automatik** měří jednoduchou proměnnou nebo polohu fázoru v komplexní rovině.
- **Automatiky** se dají v počátku zablokovat a odblokovat během výpočtu jinou **automatikou**.
- Program zobrazuje čas v diskretních krocích.
- Je-li perioda **automatik** menší než integrační krok, má integrační krok prioritu (nezkracuje se).
- Je-li interval výpisu kratší než integrační krok dx, má krok prioritu (interval výpisu=dx).
- **Zásahem** 'SAMP' se dá změnit velikost interval výpisu během výpočtu.
- **Zásahem** 'STEP' se dá změnit velikost integračního kroku během výpočtu.
- Parametrem N\_gr=5 ve vstupním souboru VYSTUP . DAT se zvolí zobrazení v komplexní rovině.
- **Zásah** 'FOUL' umožňuje simulovat zkratky v různých místech **větve**.
- Může se jednat o trojfázové kovové a obloukové zkratky nebo zkratky nesymetrické.
- Nesymetrické zkratky se vyšetřují pouze v sousledné soustavě (nehodí se pro nastavení ochran).
- Výstupní soubory RIZENI , VYSTUP a SOUBORY . DAT lze editovat textově nebo dialogově
- Uživatelské výstupní soubory umožňují archivovat časové průběhy vybraných proměnných.
- Různé typy uživatelských výstupních souborů se zadávají ve vstupním souboru SOUBORY . DAT .
- Pro jména uživatelských výstup. souborů lze použít jmenovací konvence nebo explicitní definici.
- Uživatelské výstupní soubory obsahují hlavičku a sloupce dat oddělených čárkami.

⊗ Potenciální problémy:

- ◆ Při zadání cyklických výpočtů je potřeba upravit koncový čas výpočtu Tkonc ve vstupním souboru RIZENI . DAT, který musí být větší než časy definované ve **scénáři**.
- ◆ Je-li zadaná perioda **automatik** menší než aktuální integrační krok, je nutné zásahem 'STEP' provést zmenšení integračního kroku před zapůsobení **automatiky**.
- ◆ Při volbě zobrazení v komplexní rovině je nutno definovat také rozsahy reálné a imaginární osy ve vstupním souboru VYSTUP . DAT (místo rozsahů os závisle proměnných v 1. a 2. grafu).
- ◆ Počet zadávaných kružnic nebo přímek. v komplexní rovině **grafiky** nesmí převýšit počet zadávaných **trajektorií**
- ◆ Nelze zadat zásahem 'SAMP' menší interval výpisu než je integrační krok

### 3. VÝPOČTY STŘEDNĚDOBÉ DYNAMIKY

Střednědobá dynamika je charakterizována časovým rozsahem několika desítek sekund. Z poruch se vyšetřují výpadky zdrojů (případně změna zatížení) a zkoumá se odezva soustavy z hlediska frekvenční stability (hrozba frekvenčního kolapsu).

Frekvence v elektrizační soustavě musí být udržována v úzkých mezích. Vyžaduje to bezpečný provoz zdrojů a spolehlivý chod spotřebičů. V běžném provozu je frekvence udržována činností primární a sekundární regulace.

Primární regulace je odezva výkonu turbíny na odchylky frekvence od jmenovité hodnoty. Tato odezva musí být dosažena během několika sekund a musí mít trvalý charakter (pokud odchylka trvá). Pro primární regulaci se vyhrazuje z regulačního rozsahu turbíny tzv. primární regulační rezerva. Sumární primární regulační rezerva by měla tvořit 2.5 % celkového výkonu zdrojů. Primární regulace má proporcionální charakter<sup>1</sup>, takže zanechává trvalou odchylku frekvence. Protože na odchylku frekvence reagují všechny bloky s primární regulací (decentralizovaný charakter) je tak realizována dočasná havarijní výpomoc postiženému systému podle tzv. principu solidarity. Primární regulace může být realizována:

- regulátorem výkonu (pomocí tzv. korektoru frekvence)
- proporcionálním regulátorem otáček
- kombinací předchozích v tzv. sériovém nebo paralelním uspořádání.

Sekundární regulace vyrovná saldo předávaných výkonů na nulu a vrátí frekvenci na jmenovitou hodnotu. Toto musí být uskutečněno během několika minut. Bloky zařazené do sekundární regulace (regulační bloky) mají vyhrazeno v regulačním rozsahu tzv. regulační pásmo. Sumární sekundární regulační rezerva musí pokrýt výpadek největšího bloku v soustavě. Sekundární regulace má charakter proporcionálně - integrační<sup>2</sup>, takže ustálená regulační odchylka je nulová. Sekundární regulace je realizována centrálním regulátorem podle tzv. síťových charakteristik, takže se jí zúčastňují pouze regulační bloky v postižené soustavě (kde došlo k nerovnováze mezi výrobou a spotřebou). Takto je realizován tzv. princip neintervence - každá soustava je povinná pokrývat svou výkonovou bilanci. Sekundární regulací P a f se budeme zabývat v samostatné kapitole o dlouhodobé dynamice.

Při větších odchylkách frekvence dochází:

- ke změně regulačního režimu turbín - z režimu primární regulace do režimu regulace otáček
- k odpojování vybraných bloků od sítě při vzrůstu frekvence
- ke změně čerpadlového provozu na turbínový u přečerpacích elektráren při poklesu frekvence
- k frekvenčnímu odlehčování zátěže při poklesu frekvence
- odpojení postiženého systému od ostatní soustavy.

V této kapitole se budeme zabývat simulací výše uvedených dějů (kromě sekundární regulace). Taková simulace může být například využita pro tvorbu tzv. Obranných plánů proti šíření velkých poruch. Tyto plány zajišťují, aby porucha, která vznikne v jednom systému neohrozila bezpečný provoz ostatních propojených systémů.

Pro započetí práce otevřeme projekt MIDDLE.

<sup>1</sup> říkáme také statický podle sklonu statické charakteristiky  $N_t = \text{funkce}(f)$

<sup>2</sup> říkáme také astatický

### 3.1. Zadání výpadku bloku pomocí scénáře

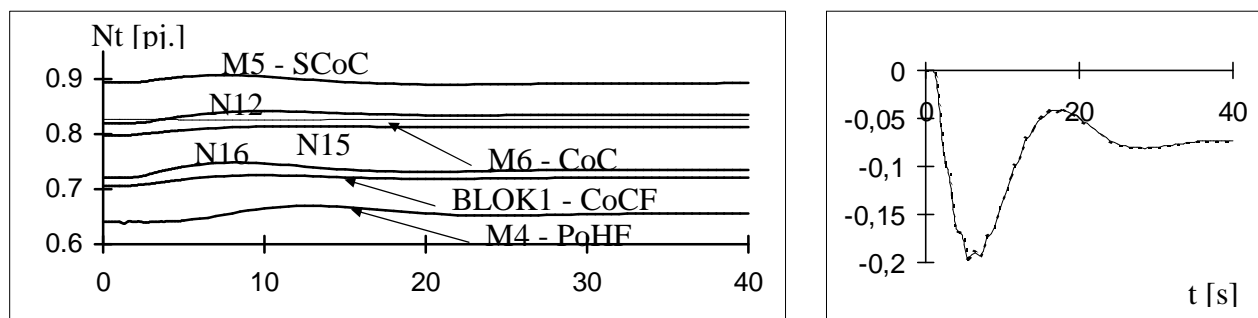
Jak již bylo řečeno slouží pro simulaci poruch scénář zadávaný v souboru SCENAR.DAT.

- Pro definici poruchy použijeme makro out\_unit, definující výpadek bloku BLOK3. Zároveň změníme v páté sekundě původní interval výpisu<sup>1</sup> na půl sekundy:

<code>\$(out_unit:unit_1=BLOK3 )</code>	<code>1.,'UNIT'</code>
<code>5.,'SAMP'</code>	<code>1</code>
<code>1</code>	<code>'unit_1',0</code>
<code>0.5,0.001</code>	
<code>10000,'END'</code>	

#### Výp. 14 Výpis vstupního souboru SCENAR.DAT pro simulaci výpadku bloku

Při vyšetřování střednědobé dynamiky nás zajímá především průběh frekvence. V programu se frekvence zobrazuje ve formě proměnné značené symbolem<sup>2</sup> 'SU' definované jako poměrná odchylka frekvence od jmenovité hodnoty (kladná při frekvenci větší než jmenovitá a záporná při menší). Také je možno vynášet poměrnou odchylku otáček bloku - skluz generátoru 'SG'. U první proměnné se zadává jméno uzlu u druhé jméno bloku. Dále nás bude zajímat průběh výkonu turbíny (symbol 'NT'), na kterém budeme vidět odezvy na změnu frekvence - činnost primární regulace. Využijeme maximální možnosti zobrazení sedmi průběhů na vybraných blocích. Tyto proměnné jsou definovány ve vstupním souboru VYSTUP.DAT. Po spuštění programu grafika ukáže:



Obr. 6 Grafika s průběhy výkonu turbín a odchylky frekvence

Povšimneme si nejdříve pravého grafu s průběhy odchylek frekvence napětí 'SU' v uzlu a skluzu generátoru 'SG' bloku. Je vidět, že obě proměnné téměř splývají, což je logické, neboť blok N16 je připojen do uzlu N16 (je zde shoda jmen, jinak ovšem záleží na uživateli, jak si pojmenuje uzly a bloky<sup>3</sup>). Je známo, že střední hodnota frekvence v synchronně propojeném systému je stejná, okamžité hodnoty však mohou oscilovat kolem střední hodnoty díky elektromechanickým kyvům jednotlivých generátorů. Vzhledem k tomu, že v běžném provozu jsou odchylky frekvence a skluz generátoru malé, jsou v grafice vynášeny v procentech, tj. 100x zvětšené proti ostatním proměnným, které jsou v grafice vynášeny v poměrných jednotkách. Jednomu dílku v grafice odpovídá 0.01 pj. tedy 500 mHz. Koncová hodnota frekvence v čase 40s tedy činí  $-0.76 \times 0.1 \times 500 = -38 \text{ mHz}$ , což přibližně odpovídá údajům okamžité hodnoty v horní části grafiky.

V levém grafu jsou časové průběhy mechanického výkonu turbín vybraných bloků. Průběhy mají odlišnou dynamiku podle toho, jak je realizována primární regulace. V úvodní kapitole byly popsány základní možnosti realizace pomocí korektoru frekvence a regulátoru otáček.

Červený průběh bloku M6 odpovídá parní turbíně bez korektoru frekvence a bez regulace otáček. Tento blok tedy udržuje zadaný výkon bez ohledu na frekvenci (tzv. klasická regulace).

Horní světlomodrý průběh bloku M5 odpovídá sériovému uspořádání hydraulického regulátoru otáček a elektronického regulátoru výkonu bez korektoru frekvence. Je vidět, že v tomto případě se nejedná o primární regulaci ve smyslu definované v úvodní kapitole, neboť odezva výkonu turbíny na odchylku frekvence je pouze dočasná. Je to způsobeno tím, že regulátor výkonu

<sup>1</sup> zadávaný ve vstupním souboru RIZENI.DAT

<sup>2</sup> Symboly proměnných jsou umístěny v katalogu SYMBOL.CAT v e vstupním podadresáři

<sup>3</sup> Jména uzlů jsou definovány v tabulce uzlů UST.DAT a jména bloků v tabulce bloků GEN.DAT

turbíny, který má proporcionálně integrační charakter (a tedy nulovou ustálenou regulační odchylku) srovná po dočasném zvýšení regulátorem otáček výkon turbíny na zadanou hodnotu.

Další případy již ukazují správnou funkci primární regulace. Spodní (zelený) průběh patří vodní turbíně bloku M4 (v režimu regulace výkonu s korektorem frekvence) - je vidět pomalejší odezva oproti ostatním blokům (parní turbíny). Je to způsobeno pomalejším otevíráním regulačního mechanismu vodní turbíny, jehož rozměry jsou řádově metry oproti řádově centimetrům zdvihu regulačních ventilů parní turbíny. V prvních okamžicích dokonce vodní turbína výkon snižuje, což je způsobeno rázem vodního sloupce po otevírání rozvodného mechanismu.

Nejrychlejší odezvu má blok N16 (tmavomodrý průběh), který pracuje v režimu paralelního uspořádání elektronického regulátoru výkonu s korektorem frekvence a regulátoru otáček. Změna frekvence se tedy v regulaci projeví dvakrát. Jednak korektor frekvence opraví zadanou hodnotu pro regulátor výkonu a jednak změnu otáček zaregistruje regulátor otáček. Výstupy obou regulátorů se sčítají a přes rychlý elektrohydraulický převodník ovládají tlak regulačního oleje.

O něco pomalejší odezvu má blok N12 (fialový průběh), který pracuje v režimu sériového uspořádání elektronického regulátoru výkonu s korektorem frekvence a hydraulického regulátoru otáček. Regulátor výkonu mění přes elektromechanický převodník zadanou hodnotu hydraulického regulátoru otáček, který vytváří tlak regulačního oleje.

Ještě pomalejší odezvu má blok BLOK1 (modrý průběh), který je řízen elektronickým regulátorem výkonu s korektorem frekvence bez regulátoru otáček.

Předchozí tři bloky měly primární regulační rezervu omezenou na  $\pm 5\%$  jmenovitého výkonu turbíny, jak je běžné pro naše parní turbíny. Vodní turbína toto omezení nemá a může tedy měnit výkon v celém regulačním rozsahu. Poslední blok N15 má primární regulační rezervu omezenou na  $\pm 2.5\%$ , takže její příspěvek v přechodné části děje je menší.

Koncové hodnoty výkonu pro regulační bloky jsou ve všech případech vyšší než počáteční ustálené hodnoty asi o 1.5%, což odpovídá stejné statické primární regulaci (5%) pro všechny bloky a požadavku trvalé odezvy výkonu turbíny na odchylku frekvence.

V popisu Obr. 6 jsou uvedeny kódy regulačního režimu turbíny. Vysvětlení je následující:

S	-funkční regulátor otáček
-	-(mezera) bez regulátoru otáček
	režim regulace turbíny
Po	- klasická regulace
Pr	- předtlaková regulace
Co	- koordinovaná regulace
Ma	- ruční řízení
Sl	- přirozený klouzavý tlak
	typ bloku
C	- konvenční s konstantním tlakem
V	- konvenční s proměnným tlakem
N	- jaderný blok
H	- vodní blok
G	- plynový blok
F	- funkční korektor frekvence
-	-(mezera) bez korektoru
XXXXX	

Uvedený kód stručně popisuje uspořádání regulace pohonu<sup>1</sup>. Program na základě zadaných modelů a typových parametrů kód identifikuje a vypíše ho do výstupního souboru PRIM\_REG.ANA v podadresáři VYST. V tomto souboru je uveden přehled všech bloků s hodnotami primární regulační rezervy, regulačního rozsahu, pracovního bodu a jmenovitých výkonů turbíny a generátoru. Dále je v souboru uveden přehled jednotlivých oblastí s celkovou hodnotou generovaného výkonu a sumární primární regulační rezervy oblasti absolutní v MW a procentní.

Během výpočtu probíhá piktogram regulačních ventilů. Tímto program signalizuje, že zapůsobila přídatná automatika, která po odpojení bloku od sítě rychle zavírá ventily turbíny aby nedošlo k nebezpečnému zvýšení otáček.

Po zvládnutí přeběhu je automaticky přepnuto do regulace otáček -viz výpis výstupním souboru AKCE.HLA:

Cas [s]	Objekt	Udalost
1.000	BLOK3	odpojen blok dle scenare
1.000	BLOK3	blok odpojen od site -> rychle zavirani ventilu
1.000	BLOK3	prepnuo do prac.regulace otacek a pripraveno na synchronizaci

### Hla. 3 Hlášení o provedení zásahů podle scénáře a činnosti přídatných automatik

<sup>1</sup> Regulace pohonu je širší pojem než regulace turbíny, neboť se obsahuje i regulaci zdroje pohonného média



### 3.2. Definice modelů a typových parametrů

Předchozí výklad byl zaměřen na analýzu dynamického chování jednotlivých typů regulace turbíny. Dynamické chování je určeno strukturou a parametry modelu:

Program MODES používá pro definici struktury a parametrů modelu tyto koncepty:

- výběr z knihovny modelů
- výběr typových parametrů z katalogu.

Knihovna obsahuje hotové modely<sup>1</sup>, pro akční člen - turbínu je možno zvolit tyto modely:

- \* 'CONS' - konstantní výkon,
- \* 'STAN' - standardní model,
- \* 'ST\_S' - zjednodušený model parní turbíny,
- \* 'ST\_A' - podrobný model parní turbíny včetně záchytných ventilů
- \* 'HYDR' - nelineární model vodní turbíny
- \* 'DIES' - model přeplňovaného vznětového motoru
- \* 'GAST' - model plynové turbíny.

Pro regulátor se používá jednotný model s drobnými úpravami pro jednotlivé typy turbín, což program identifikuje sám, takže model regulace nemusí být explicitně zadáván.

Katalog typových parametrů bloku je umístěn ve vstupním souboru TYP\_BLOK.CAT a členěn na úseky typových parametrů:

- generátoru
- budičů
- regulátoru buzení
- přídatných automatik regulátoru buzení
- turbín
- regulátorů pohonu
- přídatných automatik regulátoru pohonu
- zdrojů pohonného média - kotle nebo tlakovodního reaktoru
- asynchronního motoru
- poháněného zařízení asynchronního motoru.

Přiřazení modelů a a typových parametrů jednotlivým **blokům** se provede v databázovém vstupním souboru BLOK.DTB, který obsahuje seznam bloků s určením typu modelu a křížového odkazu do katalogu typových parametrů. Z uživatelského rozhraní je soubor dostupný příkazem **Modifikovat| Modely bloků** nebo čtvrtým tlačítkem v třetí skupině tlačítkové lišty.

☐ V textovém režimu se objeví následující výpis:

V souboru BLOK.DTB jsou data pro modelovani bloku a motoru jednotny model gen.(IND-individ.)														
'IND'														
por. nazev	Generator		Budic		Turbina				Zdroj					
C. bloku uzlu	Typ	Model	Typ	Model	Regul	Prid	Typ	Model	Regul	Prid	Stav	Typ	Model	
											Ngenx			
1 'BLOK1	'N1	'J950+T	'PARK	'CIGRDC	'DC_1	'default	'default	'M1-M5	'ST_A	'CoCF	'EHS	'ZAP	'default	'CONS'
9 'M4CERP	'N103	'V130+T	'PARK	'CIGRDC	'DC_1	'default	'default	'V_100F	'HYDR	'PuH	'A-ROT	'VYP	'Hydro	'CONS'

#### Výp. 15 Zadání modelů a typových parametrů jednotlivým blokům v databázi BLOK.DTB

Pro turbínu se zadávají čtyři sloupce. Jsou to křížové odkazy do katalogu typových parametrů turbíny, regulátoru pohonu, přídatných automatik, dále volba modelu z knihovny a stav regulátoru turbíny<sup>2</sup>. Za povšimnutí stojí, že databáze může obsahovat libovolný počet záznamů. Program si vybere potřebné podle aktuální skladby bloků (definované ve vstupním souboru GEN.DAT). Jestliže blok není v databázi nalezen, jsou mu přiřazeny standardní modely a standardní parametry ('default').

<sup>1</sup> Popis modelů je v Popisu[1]

<sup>2</sup> 'ZAP' / 'VYP' podle toho jestli je regulace zapnuta / vypnuta (ruční řízení nebo klouzavý tlak viz [1])

☒ V dialogovém režimu se otevře Editor modelů bloků.

Dialog do sebe integruje informace ze tří vstupních souborů - z **tabulky**, **databáze** a **katalogu** typových parametrů modelů **bloků**. V seznamu *Seznam Bloky* jsou uvedeny všechny **bloky** z **tabulky bloků** (GEN.DAT nebo příslušná variace chodu sítě CHODS#\_\_\_\_.GEN). Kliknutím na jméno v seznamu se vybere aktuální **blok**.

Ve čtyřech kombo boxech v rámečku *Výběr komponenty modelu* jsou uvedeny modely s knihovny přiřazené aktuálnímu **bloku**. Přiřazení modelů a typových parametrů je provedeno v základní **databázi** BLOK.DTB, případně v **modifikaci** BLOK.### (jméno příslušné **databáze** je v textovém poli *Databázový soubor*). Stlačením příslušného tlačítka se v tabulce<sup>1</sup> *Sady typových parametrů* posune šipka v levém sloupci na záznam s typovými parametry přiřazenými aktuálnímu **bloku**. Zároveň se tyto parametry zkopírují do řádku *Typové parametry vybraného bloku*.

Výběrem modelu z kombo boxu (při stlačení tlačítka ve sloupci nad kombo boxem) navolit model přiřazený aktuálnímu **bloku**. Volba se provede tlačítkem **Vyměnit model**.

Bloku je možno přiřadit jinou sadu typových parametrů. Kliknutím na levý sloupec tabulky se tam přemístí šipka, pak stiskneme tlačítko **Vyměnit parametry** a výměnu parametrů potvrdíme.

Tlačítkem **Vyřadit záznam** je možno vyřadit záznam z **databáze bloků**. Pokud **blok** není v žádné **databázi** modelů (ani základní ani **modifikaci**) je možno přidat standardní záznam tlačítkem **Přidat záznam** do databáze. Standardní záznam lze pak upravit výše popsanou volbou modelů v kombo boxech a změnou typových parametrů.

Sadu typových parametrů lze z tabulky vymazat klávesou Delete. Tento způsob se použije jen pro případ duplicitních záznamů v globálním a lokálním **katalogu**, protože záznamy mohou být používány i jinými **případy**. Kliknutím na levý sloupec tabulky se daná sada stane aktuální (řádek zmodrá). Najedeme kurzorem na řádek a klikneme. Pak stiskneme klávesu Delete. Toto vymazání je možné pro jen pro lokální **katalog**, protože sada z globálního může být používána jiným **projektem**.

Pro přehled využívání záznamů z lokálního **katalogu** pro jednotlivé **případy** (přesněji jednotlivými **modifikacemi databází**) se použije příkaz **Modifikovat| Katalogy bloků**.

☐ V textovém režimu MODMAN otevře oba katalogy lokální i globální.

☒ V dialogovém režimu se otevře dialog *Katalog typových parametrů modelů bloku*, kde v rámečku *Overview of parameters using* je seznam všech záznamů v základní **databázi** BLOK.DTB a všech **modifikacích** v **projektu**. Pohybem po tabulce sad parametrů se seznam mění. Stiskem tlačítek v rámci *Select the Section* se mění tabulka sad parametrů.

Každý řádek v katalogu je označen svým symbolickým jménem, pomocí něhož se bloku v databázi přiřadí příslušné typové parametry.

V principu tedy existují dvě metody jak změnit parametry modelu:

1. změnit hodnotu parametru v stávajícím řádku katalogu
2. přidat nový řádek typových parametrů s novým jménem a opravit křížový odkaz v databázi.

První způsob se projeví na všech **blocích**, které dané typové parametry používají. Druhý způsob se projeví pouze u dotyčného **bloku**, kterému jsou nové typové parametry přiřazeny. Doporučuje se používat druhý způsob, který je méně „destruktivní“. Proto si ho ukážeme na příkladu. Předpokládejme, že oblast Ob12 bude chtít z ekonomických důvodů snížit velikost primární regulační rezervy. Podle schématu uvedeném v následující části je v uvedené oblasti jediný blok N12, který má přiděleny typové parametry 'SCoCFs'.

<sup>1</sup> Tabulka obsahuje záznamy z globálního a lokálního **katalogu** TYP\_BLOK.CAT v pořadí jako jsou při inicializaci prohledávány a přiřazovány jednotlivým modelům z **databáze** BLOK.DTB

V textovém režimu napřed v katalogu zkopírujeme příslušný řádek na konec úseku parametrů regulátoru, přejmenujeme jej na 'SCoCF2' a přepíšeme hodnoty parametrů Nfmax a Nfmin na 2.5 -2.5. Tímto jsme vlastně omezili výstup z korektoru frekvence na  $\pm 2.5\%$ . Jelikož jsme v úseku regulátorů pohonu přidali řádek, je nutno opravit počet řádků v hlavičce úseku na 10. Úsek s typovými parametry pak bude vypadat takto:

C NTrtur-pocet typu regulátoru turbíny (max 30):																								
C																								
11																								
C Por Jm. A1 A2 TiT TiB TN TEH kT kB kSp kFr kCor kPr kFor Gen VN Změny krokuN dFr dSp dPres dp NFmax NFmin																								
C cis. [%/min] [%]																								
C NTtrturx																								
1	'default'	1	0	10	100	1	10	1	1	20	0	20	1	0.5	1	1	0.0	0.0	0.05	0.0	0.0	5	-5	
10	'SCoCF1'	1	0	10	100	1	10	1	1	20	0	20	1	0.5	1	50	50.0	0.0	0.05	0.0	0.0	2:5	-2:5	
11	'SCoCF2'	1	0	10	100	1	10	1	1	20	0	20	1	0.5	1	50	50.0	0.0	0.05	0.0	0.0	2:5	-2:5	

### Výp. 16 Přidání nového řádku s typovými parametry do úseku regulátoru pohonu

V databázi BLOK.DTB opravíme křížové odkazy pro typové parametry regulátorů pohonu bloku N12, takže příslušný řádek bude vypadat takto:

V souboru BLOK.DTB jsou data pro modelování bloku a motoru jednotny model gen.(IND-individ.)														
'IND'														
por. nazev	Generator		Budic		Turbina				Zdroj					
C. bloku uzlu	Typ	Model	Typ	Model	Regul	Prid	Typ	Model	Regul	Prid	Stav	Typ	Model	Ngenx
5 'N12'	'N12'	'default'	'PARK'	'default'	'DC_1'	'default'	'default'	'M1-M5'	'ST_S'	'SCoCF1'	'REGULI'	'ZAP'	'default'	'CONS'
6 'N15'	'N15'	'default'	'PARK'	'default'	'DC_1'	'default'	'default'	'M1-M5'	'STAN'	'SLACK'	'default'	'ZAP'	'default'	'CONS'

### Výp. 17 Zadání modelů a typových parametrů jednotlivým blokům v databázi BLOK.DTB

- ☒ V dialogovém režimu se otevře **dialog Databáze modelů bloků**:
  - ☞ v rámečku *Výběr komponenty modelu* stiskneme tlačítko **Regulátor pohonu**
  - ☞ v seznamu bloků vybereme N12
  - ☞ klikneme do posledního řádku tabulky (s hvězdičkou) a stiskneme **Přidat novou sadu**
  - ☞ po potvrzení *Ano* se do prázdného řádku zkopíruje původní sada parametrů SCoCFs
  - ☞ opravíme název na jej na SCoCF2 a přepíšeme hodnoty parametrů Nfmax a Nfmin na 2.5 -2.5
  - ☞ stiskneme **Vyměnit parametry** a po potvrzení *Ano* se modelu přiřadí nová sada SCoCF2
  - ☞ Dialog ukončíme stiskem tlačítka **OK** a potvrdíme vznik nové **modifikace** BLOK.001.
- Modifikace** BLOK.001 obsahuje pouze jeden záznam pro blok N12. Při vytváření dalších modifikací se kromě editovaného záznamu do nové modifikace zkopírují záznamy původní. Jako vedlejší produkt vzniká v podadresáři VST soubor se stejným názvem jako **katalog** a s příponou MDB. Soubor představuje databázi typových parametrů ve formátu MS ACCESS.

Je možné úpravy přeskočit o otevřít si **případ** ModifN12.

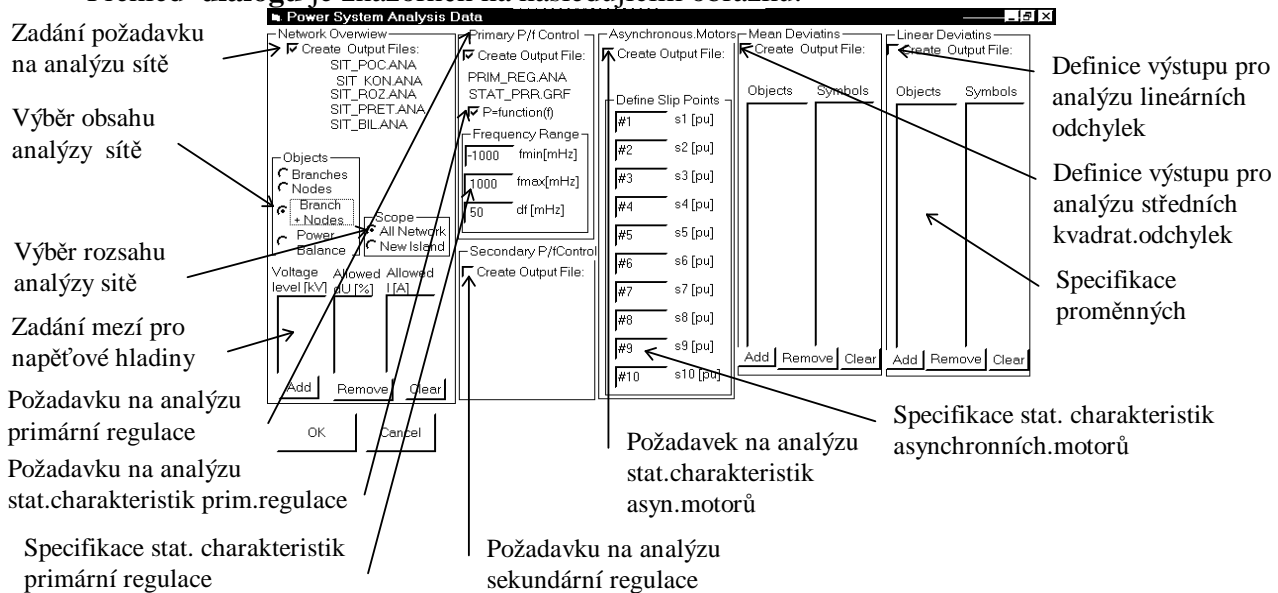
Po opětovném spuštění programu zjistíme, že maximální hodnota odchylky frekvence poklesla z 97 mHz na 100 mHz. Ustálená odchylka frekvence zůstala stejná, neboť odpovídající změna výkonu turbíny (cca 1.5 %) je pod zadaným omezením  $\pm 2.5\%$ .

Primární regulační rezerva v oblasti Obi2 poklesla na hodnotu 3.1 %, jak se čtenář může přesvědčit ve výstupním souboru PRIM\_REG.ANA (**Analýza Primární regulace f**). Požadavek na analýzu se provede příkazem **Modifikovat| Analýza**. V textovém režimu se zadá parametr Ana2 roven 1.

- ☒ V dialogovém režimu se na kartě *Primary f control* zaškrtně **Create Output Files**. Je také možno zadat požadavek na tvorbu statických charakteristik primární regulace zaškrtnutím  $P=function(f)$  a zadáním rozsahu odchylky frekvence a kroku. V podadresáři VYST pak vznikne soubor STAT\_PRR.GRF, který obsahuje závislosti výkonu uvolňovaného jednotlivými **oblastmi** při odchylkách frekvence. Tento výkon je uvolňován činnostmi korektoru frekvence regulátoru výkonu nebo regulátoru otáček, jestliže není zapnut regulátor výkonu (jedná se o trvalou rezervu

ve sloupci Static), případně regulátorem otáček v sériovém uspořádání s regulátorem výkonu bez korektoru frekvence (jedná se o dočasnou rezervu ve sloupci Dynam)<sup>1</sup>.

Přehled **dialogu** je znázorněn na následujícím obrázku:



#### Dial. 4 Přehled karet dialogu pro analýzu s významem jednotlivých prvků

Výpadek **bloku** BLOK3 s výkonem 175 MW byl zvládnut činností primární regulace, neboť deficit výkonu je menší než sumární primární regulační rezerva 372 MW. V případě, že vypadlý výkon je větší než primární regulační rezerva, primární regulace pokles frekvence nezastaví a působí jednotlivé stupně frekvenčního odlehčování. To bude popsáno v 2.díle průvodce.

Tímto jsme vyčerpali základní témata simulace střednědobé dynamiky a následuje stručné shrnutí uvedené problematiky.

Důležité body:

- Koncový čas výpočtu a interval výpisu se zadává ve vstupním souboru RIZENI.DAT
- Frekvence je v programu zobrazována ve formě odchylky od jmenovité hodnoty
- V jednom grafu na obrazovce lze zobrazit maximálně 7 časových průběhů **proměnných**
- Dynamické chování bloku je určeno strukturou a **parametry modelu**
- Pro definici **modelu** se používá **knihovna** hotových modelů a **katalog** typových parametrů
- **Knihovna** programu obsahuje pět modelů turbíny a univerzální model regulátoru pohonu
- Regulátor pohonu se skládá z regulátoru turbíny a kotle a vazeb mezi nimi
- Modely a odkazy na typové parametry se zadávají v databázi BLOK.DTB
- Typové parametry jsou definovány v **katalogu** TYP\_BLOK.

⊗ Potenciální problémy:

- ♦ Počet znaků řetězce a jména v definici makra musí mít stejný počet znaků<sup>2</sup>
- ♦ Změna typového parametru ovlivní všechny bloky, které mají dotyčné parametry přiřazené
- ♦ Kód režimu regulace<sup>3</sup> nemusí odpovídat jménu typ.parametrů regulátoru pohonu
- ♦ Pro správnou funkci tabulkového režimu (As Table) prohlížení a úpravy databází je nutné, aby řádkové komentáře (textové poznámky na konci řádku typových parametrů) v katalozích začínaly vykřičníkem.

Účelem výpočtů střednědobé dynamiky je především zkontrolovat odezvu soustavy při větších deficitech výkonu (při výpadcích bloků). Jako obranné prostředky proti poklesům frekvence slouží zejména primární regulace a frekvenční odlehčování zátěže.

Některé výpočty střednědobé dynamiky byly publikovány v [4] a [5].

<sup>1</sup> Popis regulace turbíny je [1].

<sup>2</sup> např: \$(MAKRO:Uzel\_1=N1 )

<sup>3</sup> zapisovaný ve výst.souboru PRIM\_REG.ANA

## 4. VÝPOČTY DLOUHODOBÉ DYNAMIKY

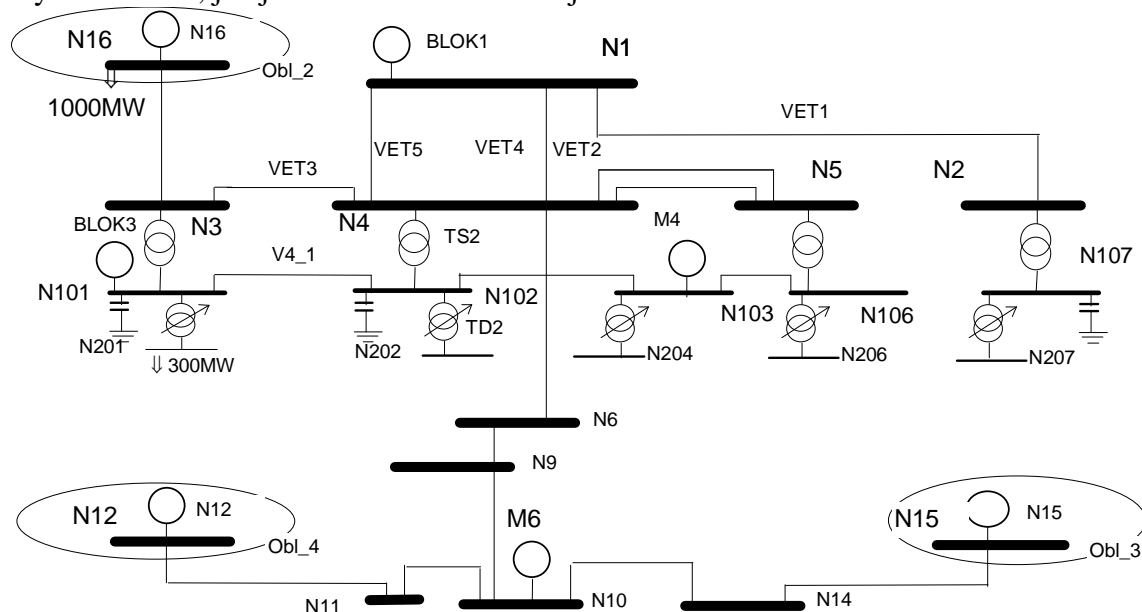
Dlouhodobá dynamika je charakterizována časovým rozsahem jednotek až desítek minut. Je možné vyšetřovat jednak hrozbu frekvenčního kolapsu a jednak hrozbu napětového kolapsu.

Z hlediska frekvenčního kolapsu je dlouhodobá dynamika pokračování střednědobé dynamiky, která byla popsána v předchozí kapitole. Jak již víme primární regulace zanechává trvalou odchylku frekvence. Její odstranění je úkolem sekundární regulace P a f. Tuto funkci plní centrální regulátory v jednotlivých oblastech. Vstupní data je potřeba doplnit o údaje regulátorů.

Z hlediska napětového kolapsu ovlivňují tento děj zejména omezovače rotorového proudu<sup>1</sup> a přepínání odboček regulačních traf. Obojí mechanismy lze v dlouhodobé dynamice simulovat.

### 4.1. Simulace dlouhodobé dynamiky z hlediska frekvence

Známým způsobem otevřeme projekt nazvaný LONG, který obsahuje připravená vstupní data pro simulaci výpadku bloku v dlouhodobé dynamice. Modelovaná soustava je rozdělena do čtyřech oblastí, jak je naznačeno na následujícím schématu:



Sch. 2 Rozdělení modelované soustavy do oblastí

Většina uzlů spadá do oblasti Obl\_1, která není na schématu explicitně vyznačena. Zbylé tři oblasti jsou označeny elipsami s názvem oblasti. Příslušnost uzlu k oblasti je definovaná v souboru UST.DAT číslem oblasti. Jednotlivé oblasti jsou pak definovány jménem v samostatném úseku stejného souboru.

Oproti předchozímu projektu střednědobé dynamiky se provedou změny časových úsecích :

- koncový čas výpočtu je  $T_{kon} = 600s$  (v souboru RIZENI.DAT)
- interval výpisu je  $Int = 1s$  (v souboru RIZENI.DAT)
- zablokování integračního kroku na hodnotě  $0.1s^2$  (v souboru RIZENI.DAT)
- zadání výpadku zásahem UNIT v 10s (v souboru SCENAR.DAT)
- prodloužení intervalu výpisu v 40s na 5s (v souboru SCENAR.DAT)

tyto změny jsou již předem připraveny v případě ZAKLADNI, takže jej můžeme otevřít a spustit výpočet. Grafy nám neukáží nic nového, po odeznění primárního regulačního děje je průběh stabilní s ustálenou odchylkou  $\Delta f = 37 \text{ mHz}$ .

Přistoupíme k simulaci činnosti sekundární regulace. Všechny vstupní data jsou připraveny ve vstupním souboru AUTSEK.DAT, stačí sekundární regulaci odblokovat pomocí scénáře zásahem 'AUTO', kde se jako objekt uvede 'LFC'<sup>3</sup>:

<sup>1</sup> Jsou to přídatné automatiky regulátoru buzení - viz Popis [1]

<sup>2</sup> Proveďte se volbou  $Dx = Dx_{max} = 0.1$

<sup>3</sup> z angličtiny „Load Frequency Control“

Provedeme příkaz **Modifikovat| Scénář**.

☐ V textovém režimu upravíme soubor do následující podoby:

```
0,'AUTO'
1
'LFC',1
$(out_unit:unit_1=BLOK3)
40.,'SAMP'
1
0.5,0.001
10000,'END'
```

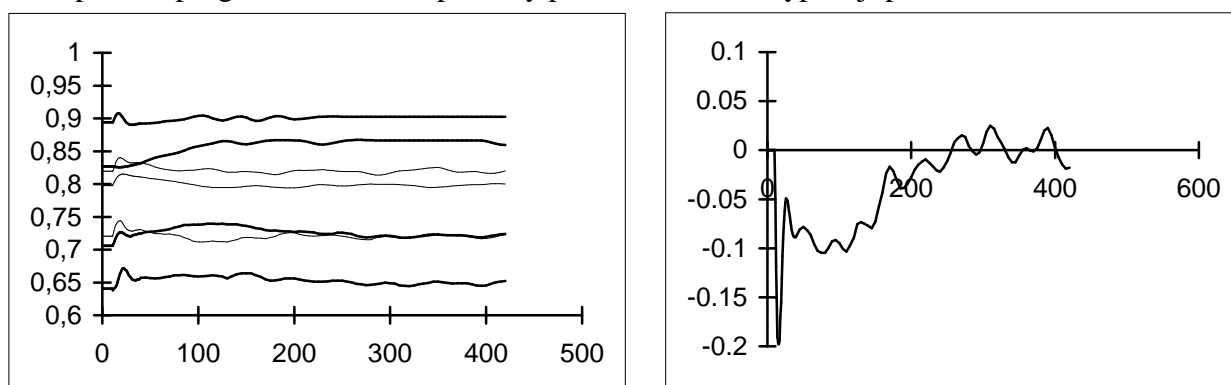
```
10.,'UNIT'
1
'unit_1',0
```

### Výp. 18 Výpis vstupního souboru SCENAR.DAT s odblokováním sekundární regulace P/f

☑ V dialogovém režimu stiskneme **AddEvent**, vybereme *Automatiku*, zadáme čas 0 a stiskneme tlačítko **AddObject**. V okně *Object Selection* vybereme *ON*, stiskneme dvakrát dvojici **Add** a **Escape** a scénář uložíme tlačítkem **OK** doporučeným způsobem (vytvoří se **variac** SCENAR.002).

Je možno otevřít rovnou **případ AktivLFC**.

Po spuštění programu se časové průběhy poněkud změní. Vypadají přibližně takto:



**Obr. 7 Grafika s průběhy výkonu turbín a odchylky frekvence**

Povšimneme si nejdříve pravého grafu s průběhy odchylek frekvence napětí 'SU'. Je vidět vliv činnosti centrálních regulátorů, které podle tzv. síťových charakteristik<sup>1</sup> vyregulují odchylku frekvence. V našem případě se podařilo dostat odchylku frekvence na -4 mHz a to během přibližně 10 minut při srovnání výkonové bilance oblastí (viz salda ve čtvrtém grafu), což lze považovat za přijatelný výsledek.

Průběhy výkonu turbíny jsou odlišné podle toho jestli blok je v postižené části systému, nebo naopak v části poskytující havarijní výpomoc (činností primární regulace) podle principu solidarity. V prvním případě se jedná o bloky BLOK1, M4, M5 a M6 kreslené silnější čarou (záměrně byly vybrány regulační bloky zapojené do sekundární regulace P a f). Tyto bloky na sebe přebírají zatížení vypadlého bloku, neboť centrální regulátor postižené oblasti jim zvyšuje zadanou hodnotu výkonu. Naopak sekundární regulace nepostižených oblastí by neměla reagovat (při správném seřízení - zesílení odchylky frekvence přesně odpovídá výkonovému číslu oblasti) a nechat postiženou oblast vyregulovat saldo<sup>2</sup> i odchylku frekvence. Výkon turbíny se mění pouze činností primární regulace a jelikož odchylka frekvence je odstraněna, vrátí se výkon turbín bloků nepostižené oblasti na původní hodnotu před poruchou. Fialový, žlutý a tmavomodrý průběh výkonů turbín se skutečně vrací ke své počáteční hodnotě, takže sekundární regulace funguje dobře podle již zmíněnému principu neintervence.<sup>3</sup>

Pokud nás zajímá další průběhy např. toky výkonů profilů a salda oblastí, je možné v grafice zobrazit další dva grafy volbou parametru N\_gr=4 ve výstupním souboru VYSTUP.DAT, kde jsou již připraveny potřebná vstupní data. Po spuštění výpočtu se na spodním levém grafu zobrazí sumární hodnoty toků výkonů hraničními vedeními mezi oblastmi. Program umožňuje zadat tzv.

<sup>1</sup> Regulátor vyhodnocuje regulační odchylku  $ACE = Saldo + K_{sys} \cdot \Delta f$ , kde  $K_{sys}$  odpovídá výkonovému číslu oblasti

<sup>2</sup> Rozdíl plánovaného a skutečného exportu/importu výkonu dané oblasti

<sup>3</sup> Rovnováha výkonu mezi spotřebou a výrobou si zajišťuje každá soustava sama

profily<sup>1</sup>, které jsou předdefinovány ve vstupním souboru KONT.DAT. Toky profilů současně znamenají export (kladné hodnoty) a importy výkonu. V pravém dolním grafu jsou pak vidět odchylky exportů-importů od zadané hodnoty<sup>2</sup>, tzv. salda. Je vidět správná funkce sekundární regulace, která salda vyregulovala.

Činnost sekundární regulace lze zkontrolovat i v uživatelských výstupních souborech typu 'LFC', u kterých je nutné zadat jméno oblasti. V našem případě máme definovány dva tyto soubory (nacházejí se ve výstupním adresáři VYST se standardními jmény Jm\_Obl.LFC). V nich jsou vynášeny hodnoty salda, filtrované regulační odchylky (ACEf) a zadávané hodnoty jednotlivým regulačním blokům.

Souhrnné informace o sekundární regulaci P a f je možno zadat ve vstupním souboru ANAL.DAT, který je z uživatelského rozhraní dostupný příkazem **Modifikovat** | **Analýza**. V dialogovém režimu se zaškrtně požadavek na analýzu sekundární regulace (viz **Chyba! Nenalezen zdroj odkazů.**), V textovém režimu se zadá Ana4=1 v úseku sekundární regulace, viz následující výpis:

```
Analyza sekundarni regulace s vystupem do souboru SEK_REG.ANA
Ana4-priznak analyzy {NE=0, ANO=1;}
```

1

### Výp. 19 Zadání volby analýzy sekundární regulace P a f jednotlivých oblastí

Výstupní soubor SEK\_REG.ANA je dostupný příkazem **Analýza** | **Sekundární regulace** obsahuje pro každou oblast:

- přehled přímo řízených regulačních bloků s jejich regulačními rozsahy a sumární rezervou
- výpis regulačních bloků řízených směrnými hodnotami Y1, Y3 a jejich sumární regulační rezervu

Takto lze jednoduše kontrolovat sumární sekundární regulační rezervu, která má mít pro každou soustavu dostatečnou velikost pro pokrytí výpadku největšího bloku.

Tímto jsme vyčerpali základní témata simulace dlouhodobé dynamiky.

#### 🔍 Důležité body:

- Pro zadání centrálních regulátorů P a f je nutné definovat jednotlivé oblasti a profily
  - Oblasti se definují v úseku oblastí ve vstupním souboru UST.DAT
  - Profily se definují ve vstupním souboru KONT.DAT
  - Proměnné sekundární regulace P a f lze zjistit v uživatelském výstupním souboru typu 'LFC'
  - Souhrnné informace o sekundární regulaci se vypisují do výstupním souboru SEK\_REG.ANA
  - Výstupní soubor SEK\_REG.ANA vzniká po volbě Ana4=1 ve vstupním souboru ANAL.DAT
  - Program UST automaticky generuje úsek oblastí a profilů<sup>3</sup> ve vstupním souboru UST.DAT
  - Tyto oblasti a profily mají standardní jména Obli a Pr\_i-k, kde i a k jsou čísla oblastí
  - Čísla oblastí jsou definovány v tabulce uzlů ve vstupním souboru UST.DAT
  - Blok lze vyřadit v počátečním stavu z regulace parametrem Stav<0 v souboru AUTSEK.DAT
  - Blok lze zařadit do regulace během výpočtu zásahem LFCB
- ⊗ Potenciální problémy:
- ◆ Činnost sekundární regulace P a f se odblokuje pomocí scénáře zásahem 'AUTO'
  - ◆ Regulační pásma a parametry omezovače rychlosti zatěžování lze zadávat alternativně pro jeden fyzický blok pro Stav=1/-1 (pokud daný blok zadáván v tabulce bloků počtem fyzických bloků) nebo jako sumární hodnoty pro Stav=0/-2. Tento parametr se zadává v souboru AUTSEK.DAT

Ukolem výpočtů dlouhodobé dynamiky z hlediska frekvenčního je:

- kontrola koordinace primární a sekundární regulace
- kontrola sekundární regulační rezervy z hlediska dynamiky - včasné odstranění odchylek salda a frekvence.

Některé výpočty dlouhodobé dynamiky byly publikovány v [4] a [5]

<sup>1</sup> Několik větví tvoří novou entitu (objekt) zvanou profil

<sup>2</sup> Tato veličina se zadává ve vstupním souboru AUTSEK.DAT parametrem Plan

<sup>3</sup> Jedná se o profily mezi jednotlivými oblastmi, které program identifikuje pomocí kontroly topologie

## OBSAH

<b>1. ÚVOD.....</b>	<b>1</b>
1.1. TISKOVÁ KONVENCE .....	2
1.1.1. Typy písma .....	2
1.1.2. Názvy titulků .....	2
1.1.3. Grafické zvýraznění.....	2
1.1.4. Nadpisy.....	2
1.1.5. Popis příkazů uživatelského rozhraní .....	2
1.2. POUŽITÉ POJMY .....	3
1.3. PRÁCE V UŽIVATELSKÉM ROZHŘANÍ MODMAN .....	4
<b>2. VÝPOČTY KRÁTKODOBÉ STABILITY.....</b>	<b>6</b>
2.1. ZADÁNÍ ZKRATU POMOCÍ SCÉNÁŘE .....	7
2.2. ODSTRANĚNÍ ZKRATU PODLE SCÉNÁŘE .....	8
2.3. NALEZENÍ MEZNÍ DOBY TRVÁNÍ ZKRATU POMOCÍ NÁVRATU .....	12
2.4. POUŽITÍ AUTOMATIKY PRO ODPOJENÍ ZKRATU .....	14
2.5. MODELOVÁNÍ ČASU.....	15
2.6. ZOBRAZENÍ V KARTÉZSKÉ SOUŘADNÉ ROVINĚ (KR) .....	16
2.7. SIMULACE RŮZNÝCH TYPŮ ZKRATŮ.....	17
2.8. ARCHIVACE ČASOVÝCH PRŮBĚHŮ PROMĚNNÝCH.....	18
<b>3. VÝPOČTY STŘEDNĚDOBÉ DYNAMIKY .....</b>	<b>22</b>
3.1. ZADÁNÍ VÝPADKU BLOKU POMOCÍ SCÉNÁŘE .....	23
3.2. DEFINICE MODELŮ A TYPOVÝCH PARAMETRŮ .....	25
<b>4. VÝPOČTY DLOUHODOBÉ DYNAMIKY .....</b>	<b>29</b>
4.1. SIMULACE DLOUHODOBÉ DYNAMIKY Z HLEDISKA FREKVENCE.....	29