VŠB TECHNICKÁ UNIVERZITA OSTRAVA | FAKULTA MATERIÁLOVĚ TECHNOLOGICKÁ

Metody popisu systémů

Autor:

Zora Koštialová Jančíková

Ostrava, 2018

Název: Metody popisu systémů

Katedra: Katedra automatizace a počítačové techniky v průmyslu

Autoři: Zora Koštialová Jančíková

Místo rok, vydání: Ostrava, 2017

Počet stran: 79

Vydala: Vysoká škola báňská – Technická univerzita Ostrava

Neprodejné

Vysoká škola báňská – Technická univerzita Ostrava

METODY POPISU SYSTÉMŮ

učební text

Zora Koštialová Jančíková

Ostrava 2017

OBSAH

1. SYSTÉM A JEHO OKOLÍ

Čas ke studiu: 1 hodina

Cíl Po prostudování této kapitoly budete umět

- definovat a popsat základní pojmy teorie systému
- popsat chování systému, jeho stav a vlastnosti
- provést klasifikaci systémů podle různých hledisek

Výklad

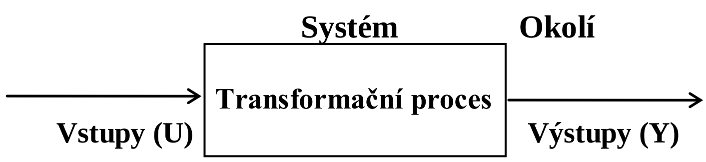
Základní pojmy

Systém je uspořádanou množinou *prvků*, mezi nimiž působí vzájemné *vazby (vztahy, relace)*, v jejichž důsledku je docilováno takového chování celku vůči okolí, které není dosažitelné působením pouhého souboru jeho vzájemně neprovázaných prvků.[1]

Za systém můžeme považovat:

- reálný objekt (společnost, počítač)
- projekt reálného objektu
- proces nebo komplex procesů (technologický proces)
- problém nebo komplex problémů
- soubor informačních, regulačních a řídících aktivit, které se vztahují k jistému reálnému problému, jeho projektu (komunikační systém, řídící systém)
- abstraktní myšlenkovou konstrukci, výrokovou konstrukci a konstrukci matematických výrazů, která je založena na reálném objektu nebo která je vytvářena bez přímého vztahu k tomuto objektu, procesu, problému

Schematická představa *působení okolí na systém i systému na své okolí* je zachycena na obr. 1. Systém transformuje *vstupní podněty (U)* z okolí na *výstupní působení (Y)*, představující reakci systému na tyto vstupní podněty. Jeho *transformační proces* je přitom určen jak vlastnostmi chování jednotlivých prvků systému, tak i jejich vzájemným uspořádáním, kterým je vymezen charakter vzájemných vazeb mezi prvky systému.



Obr. 1 Působení okolí na systém a systému na své okolí

Prvek - z nějakého hlediska dále nedělitelná část celku. Hledisku, které definuje dále nedělitelný prvek, říkáme *rozlišovací úroveň*. Nedělitelnost prvku je relativní. Z jiného hlediska (rozlišovací úrovně) může být prvek dále dělitelný. Rozlišovací úroveň se zvyšuje *dekompozicí* systému na jednodušší prvky. Zvýšíme-li rozlišovací úroveň, může se dřívější prvek stát systémem, když se v něm diferencují prvky vyšší rozlišovací úrovně a objevují se vazby mezi nimi.

Ve vztahu k okolí systému rozlišujeme prvky *vnitřní* a *hraniční*. Hraniční prvky mohou být jednak *vstupní* a jednak *výstupní*. Vstupní prvky jsou takové, jejichž vazby směřují pouze k prvkům systému a v nichž naopak vazby s prvky okolí končí. Výstupní prvky jsou takové, v nichž vazby s prvky systému končí a z nichž vazby na prvky okolí vycházejí.

Vazba systému - způsob spojení mezi prvky systému nebo mezi prvkem systému a prvkem jeho okolí. Vazby mezi prvky tvořícími systém mění pouhou množinu prvků na souvislý celek, jehož vlastnosti jsou dány jak vlastnostmi jednotlivých prvků, tak charakterem vazeb mezi nimi.

Typy vazeb:

z hlediska vztahu k okolí systému:

- *vnitřní* - spojuje prvky systému mezi sebou
- *vnější* - spojuje hraniční prvek systému s okolím

z hlediska uspořádání prvků:

- *sériová* - uspořádání prvků za sebou
- *paralelní* - uspořádání prvků vedle sebe, souběžně
- *zpětná* - spojení mezi výstupem a vstupem téhož prvku, subsystému nebo systému, které způsobuje, že vstup je závislý na výstupu; může být pozitivní nebo negativní.

z hlediska ohodnocení parametry:

parametry vazeb - kvantitativní znaky vazeb (např. násobnost vazby, časový interval trvání vazby, vzdálenost apod.)

- *jednoparametrická*
- *víceparametrická*

Okolí systému - účelově definovaná množina prvků, které nejsou prvky daného systému, avšak vykazují k němu vazby, které jsou pro daný účel významné. V okolí nás nezajímají vztahy mezi jeho prvky. Systém a okolí na sebe vzájemně působí, jsou ve vzájemné interakci. Při zkoumání interakce systému a okolí definujeme následující pojmy: *vstup, výstup systému, uzavřený a otevřený systém.*

Vstup systému (U) - množina vazeb nebo proměnných, jejichž prostřednictvím působí okolí na systém.

Výstup systému (Y) - množina vazeb nebo proměnných, jejichž prostřednictvím systém působí na okolí.

Uzavřený systém - systém, který nemá vstup ani výstup

Otevřený systém - systém, který má aspoň jeden vstup nebo výstup

Absolutně uzavřený systém je zvláštním případem, který ve skutečnosti neexistuje. Nejčastějším druhem systému jsou *relativně uzavřené systémy*, které mají některé vazby s okolím, přičemž je přesně vymezeno hledisko, z kterého je uzavřenost míněna (např. z hlediska výměny látkové, energetické nebo informační).

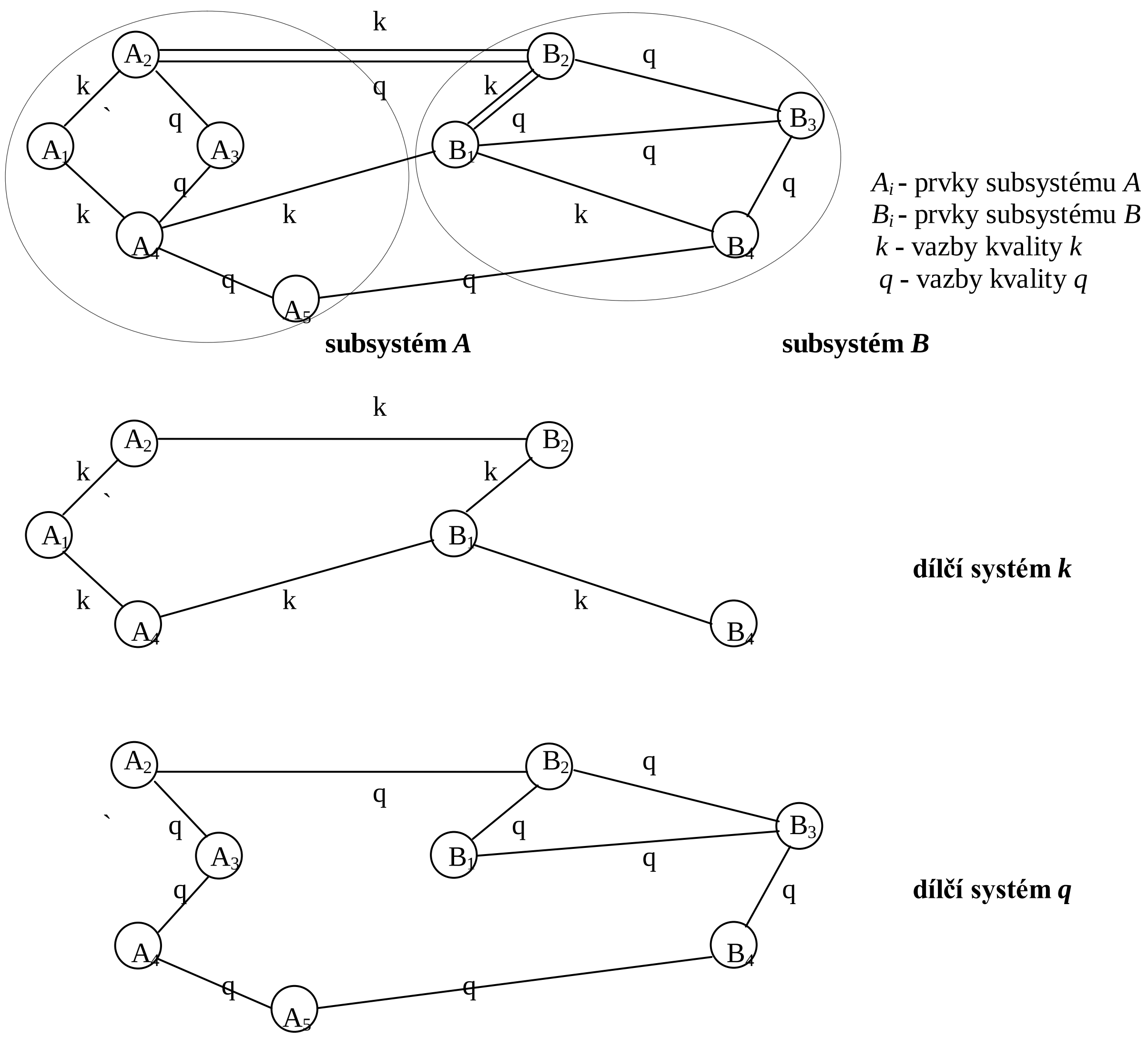
U otevřených systémů rozeznáváme jejich *podněty* a *odezvy*.

Podnět (stimul) -stav veličin vstupních proměnných, který charakterizuje dané působení okolí na systém v určitém časovém okamžiku

Odezva - stav veličin výstupních proměnných charakterizující působení systému na okolí vyvolané podnětem na vstupu systému.

Doba odezvy (časové zpoždění) - čas, který uplyne od okamžiku objevení se podnětu na vstupu systému do okamžiku objevení se k němu příslušné odezvy na výstupu systému.

V některých případech je účelné rozlišovat v systému jeho části, jejichž prvky vykazují významnější vazby mezi sebou než k jiným prvkům systému, a tvoří tedy uvnitř systému relativně samostatné celky. Hovoříme o *subsystémech* a *dílčích systémech*. Vztah mezi systémem, subsystémy a dílčími systémy je zřejmý z obr. 2:



Obr. 2 Vztah mezi systémem, subsystémy a dílčími systémy.

Za ***subsystémy*** považujeme části systému tvořené prvky téže kvality a vazbami, které tyto prvky spojují.

Za ***dílčí systémy*** považujeme řezy systému tvořené vazbami téže kvality a prvky, které tyto vazby spojují.

Subsystémy a dílčí systémy jsou tedy podmnožiny prvků a vazeb, které jsou z nějakého důvodu vyčleněny ze systému a jsou chápány buď jako nový systém nebo jako prvek.

Pro vymezení subsystému jsou rozhodující prvky začleněné do daného subsystému. Subsystém obsahuje některé prvky systému a všechny vazby definované mezi prvky začleněnými do téhož subsystému.

Pro vymezení dílčího systému je rozhodující kvalita (typ) vazeb začleněných do tohoto dílčího systému. Dílčí systém obsahuje všechny vazby dané kvality (daného typu) v systému a všechny prvky, které jsou těmito vazbami spojovány. Pokud do prvku vstupuje nebo z něj vystupuje více vazeb různých kvalit, budou vazby různých kvalit spadat do různých dílčích systémů a prvek může být současně prvkem několika dílčích systémů. V tomto prvku může docházet k překrývání struktur. Zatímco subsystémy téhož systému bývají zpravidla disjunktní (nemají žádný společný prvek), pro dílčí systémy téhož systému totéž tvrzení neplatí.

Na vysvětlení:

Vztah subsystémů a dílčích systémů lze dobře demonstrovat na lidském těle. Zatímco např. hlava, srdce, plíce, žaludek atd. mohou být považovány za subsystémy, tvoří např. nervová soustava, krevní oběh, lymfatický oběh apod. zcela zřejmé dílčí systémy.

Struktura systému - množina prvků a vazeb mezi nimi včetně jejich uspořádanosti a organizace.

Uspořádanost je pouhé vzájemné přiřazení prvků systému na základě vazeb.

Organizace předpokládá nejen uspořádanost prvků a vazeb systému, ale zahrnuje také určité specifické vzájemné vztahy mezi částmi systému, tj. mezi jeho subsystémy, dílčími systémy a hierarchickými úrovněmi.

Hierarchie je asymetrický vztah mezi dvěma prvky systému, ve kterém zdroj mezi nimi existující vazby vystupuje vůči jejímu spotřebiteli jako nadřízený a jednoznačně tak vymezuje chování spotřebitele vazby.

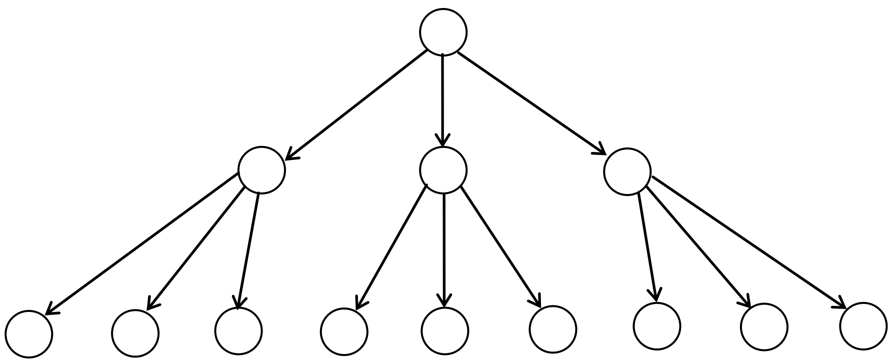
Hierarchická struktura - je taková struktura, kdy každý prvek patřící do určité úrovně je nadřazen prvku patřícímu do jiné hierarchicky nižší úrovně, přičemž každý z prvků této struktury nemůže patřit současně do více úrovní.

Vazby v hierarchické struktuře:
1) *vazby vertikální* - vazby mezi prvky různých úrovní
2) *vazby horizontální* - vazby mezi prvky téže úrovně

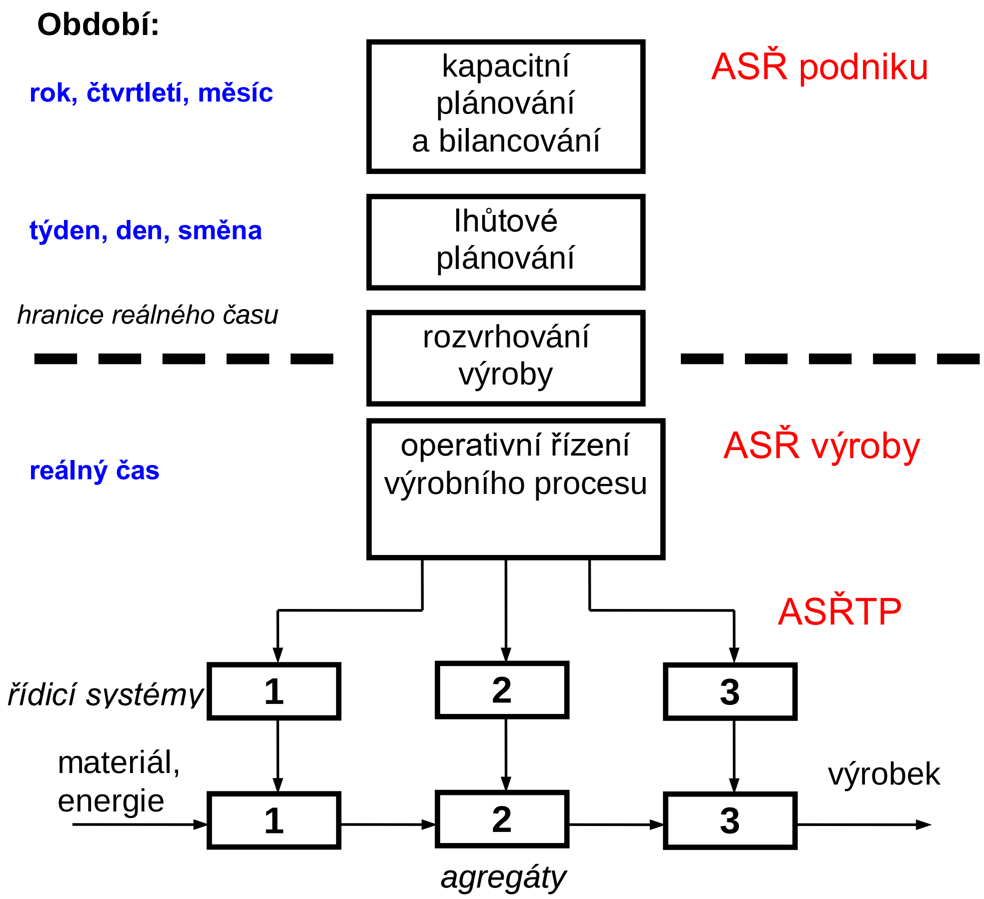
Hierarchické struktury s výlučně vertikálními vazbami představují strukturu tvaru *strom* (obr. 2).

Chování prvků na podřízené úrovni hierarchické struktury je iniciováno *povely* od prvků nadřízených a vazba od podřízeného prvku k prvku nadřízenému nepředstavuje spojení pro přenos povelů, ale slouží pouze ke *zpětnovazebnímu toku informací* o způsobu splnění přijatých povelů.

Hierarchické uspořádání systémů je v praxi zcela běžné a výjimkou je v reálném světě spíše existence systémů bez hierarchicky uspořádaných vnitřních vazeb. Příkladem hierarchického uspořádání objektivní reality světa je např. skladba krystalů. V uzlech jejich mřížky jsou molekuly složené z atomů, jejich vnitřní struktura je vymezena uspořádáním elementárních částic hmoty. Jiným příkladem hierarchie je uspořádání jednotek armády nebo hierarchická struktura automatizovaných systémů řízení technologických procesů, výroby a podniku jako celku: ASŘTP se realizuje na úrovni dílny nebo provozu a jsou zabezpečeny vazby na ASŘ výroby a ASŘ podniku podle obr. 4.



Obr. 3 Hierarchická struktura tvaru stromu



Obr. 4. Hierarchická struktura automatizovaného systému řízení v podniku

Chování systému, jeho stav a vlastnosti

Systém se v každém časovém okamžiku nachází v určitém *stavu*.

Stav systému (Z) je soubor okamžitých hodnot všech veličin systému, které lze v daném časovém okamžiku u systému rozpoznat a které spolu se znalostí vstupů systému určují jeho výstupy.

Typické stavy systému:

- *rovnovážný stav*: pro konstantní vstup **U** je výstup **Y** také konstantní

- *rovnovážný oscilační stav*: pro konstantní vstup **U** je výstup **Y** periodickou funkcí

- *nerovnovážný stav*: pro konstantní vstup **U** je výstup **Y** neperiodickou funkcí

Nerovnovážný stav může být trvalý, pak jde o *nestabilitu systému* nebo dočasný, kdy jde o *přechodový děj* v systému po vychýlení systému z rovnovážného stavu změnou vstupní veličiny nebo poruchou.

Stabilita systému je jev, kdy po vychýlení systému z rovnovážného stavu změnou vstupu nebo poruchou dochází u systému v konečném čase k novému rovnovážnému stavu. Často je návrat do původního stavu nemožný, protože se změnily podmínky, v nichž systém existuje. Pak si systém může najít stav odchylný od výchozího stavu, který je rovněž stabilní.

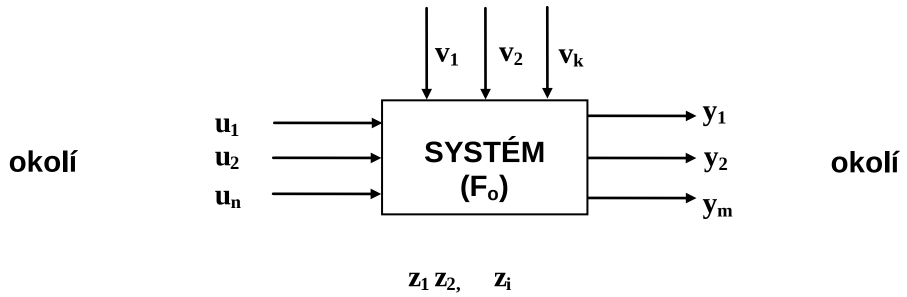
Chování systému je odezva na vstupní podnět v daném časovém intervalu. Chování systému závisí na jeho vlastnostech.

Vlastnosti systému jsou definovány jako jakákoliv podobnost v přechodech systému z jednoho stavu do stavu následujícího za známých podnětů a omezujících podmínek.

Chování systému je v podstatě projevem určitých funkcí systému. ***Funkce systému*** je výrazem časově proměnného vývoje transformačního procesu při převodu vstupního působení podnětů (stimulů) z okolí na požadované výstupní reakce (systém přechází z počátečního stavu do stavu cílového).

Transformace (probíhající v systémech) - jsou to způsoby přeměn *podnětů* prvku, subsystému nebo systému (U) na *reakce* (Y).

Schematicky můžeme tento vztah vyjádřit:



Obr. 5 Vztahy systému a okolí

Pro vektory *U*, *V*, *Y* platí:

$U = (u_1, u_2, \ldots\ldots. u_n)$ měřitelné vstupy

$Y = (y_1, y_2, \ldots\ldots. y_m)$ měřitelné výstupy

$V = (v_1, v_2, \ldots\ldots. v_k)$ .. poruchové vstupy

$Z = (z_1, z_2, \ldots\ldots z_i)$ stav systému

Operátor transformace systému (Fo) - souhrn pravidel, podle kterých se každému vstupnímu vektoru systému (U) a vektoru stavu systému (Z) přiřazuje výstupní vektor systému (Y).

Klasifikace systémů podle různých hledisek

1) *z hlediska vztahu k okolí*

- uzavřený systém - nemá vstup ani výstup
- otevřený systém - má aspoň jeden vstup nebo výstup

Pro *uzavřený systém* platí princip ekvifinality: cílový stav uzavřeného procesu je jednoznačně vymezen jeho stavem výchozím a průběhem funkcí transformačního procesu, tzn., že pokud nedojde ke změně výchozího stavu systému anebo ke změně vlastností prvků systému vymezujících funkce transformačního procesu, pak je systém vždy převeden do jednoho a téhož cílového stavu. Na platnosti tohoto principu ekvifinality jsou založeny možnosti opakovatelnosti experimentů a technických řešení výrobních procesů v různých systémech, pokud je zabezpečena uzavřenost systému. Tento princip neplatí pro *otevřené systémy*. Ty mohou dosáhnout stejného cílového stavu z různých výchozích stavů odlišnými způsoby provádění transformačních funkcí. To je hlavní příčinou obtížné opakovatelnosti výsledků biologických nebo sociálních experimentů i za stejných nebo jen málo odlišných podmínek.

2) *z hlediska zákonitostí vymezujících průběh funkcí systému*

- deterministické systémy - zákonitosti (hodnoty proměnných) vymezující chování systému jsou jednoznačně určeny (např. logické obvody)
- stochastické systémy - funkce systému jsou popisovány zákonitostmi pravděpodobnostními (proměnné se chovají náhodně), tzn., že chování systému může mít při týchž podnětech a témže stavu více variant, a to každou s určitou pravděpodobností. (např. hrací kostky, poruchy)
- neurčité (fuzzy, rozmazané) systémy - jejich funkce nelze vyjádřit žádnou zákonitostí (např. relace málo, dostatečně, mnoho...)

V prvních dvou případech se funkce systému řídí určitými zákonitostmi. Ve skutečnosti neexistuje absolutně deterministický systém, avšak stochastičnost některých systémů je tak malá, že je pokládáme za deterministické (např. šum systému je menší než rozlišovací schopnost měřicího přístroje). V případě neurčitých systémů nelze o zákonitosti v jejich chování hovořit a k jejich popisu je používáno aparátu teorie tzv. fuzzy množin.

3) *z hlediska reálné existence systémů*

- reálné systémy - objektivně existují (např. pec)
- abstraktní systémy - představované imaginárními prvky (např. matematické modely systému na počítači)

4) *z hlediska vztahu k času*

- statické systémy - stav systému se s časem nemění (např. budova)
- dynamické systémy - stav systému se s časem mění (např. zapnutá žehlička)

5) *z hlediska změn chování v čase*

- stacionární systémy (časově invariantní) - chování (stav) systému je s časem neproměnné, jeho chování při daném vstupu nezávisí na časovém intervalu, ve kterém se realizuje; parametry matematického popisu systému (např. diferenciální rovnice) jsou konstanty (např. stejně zatížený elektromotor v okolí pracovního bodu)

- nestacionární systémy (časově variantní) - chování systému je časově proměnné, jeho chování při daném vstupu závisí na časovém intervalu, ve kterém se realizuje (např. elektromotor mimo okolí pracovního bodu), parametry matematického popisu nejsou konstanty.

6) *z hlediska spojitosti veličin systému v čase*

- spojité systémy - všechny veličiny systému se mění s časem spojitě (např. teplota v peci)
- diskrétní systémy - všechny veličiny se nemění s časem spojitě, ale mají diskrétní skoky (např. veličiny v počítači - počet bitů, stav populace)
- hybridní systémy - některé veličiny jsou spojité, jiné nespojité v čase

Někdy se spojité systémy úmyslně převádějí na diskrétní nebo jsou jako diskrétní interpretovány (měření v určitých časech měřící ústřednou nebo řízení počítačem)

7) *z hlediska, zda systém dosahuje ustáleného stavu či nikoli*

- proporcionální (statické) systémy - dosahují ustáleného stavu (např. teplota v peci, žehlička)
- integrační (astatické) systémy - nedosahují ustáleného stavu (např. nádrž s přítokem bez odtoku)

8) *z hlediska charakteru matematického popisu*

- lineární systémy - všechny prvky matematického popisu systému jsou lineární operace (sčítání, odčítání, násobení konstantou, integrace, derivace)
- nelineární systémy - aspoň jedna operace matematického popisu systému je nelineární (násobení, dělení, mocniny, goniometrické funkce)

Při matematickém popisu reálných zařízení zpravidla obdržíme nelineární matematické vztahy, které značně komplikují další využití výsledků. Pro malé změny vstupních a výstupních signálů můžeme předpokládat, že vztah mezi nimi je lineární, tj. vyjádřitelný lineárními diferenciálními rovnicemi. Pro transformaci nelineárních systémů na lineární používáme postup nazvaný *linearizace*.

9) *z hlediska schopnosti systémů vyšší úrovně*

- optimální systémy - systémy, které mají z určitého hlediska optimální vlastnosti; při optimalizaci systému musíme proto definovat kritérium optimality (např. optimálně seřízený motor - kritérium: co nejnižší spotřeba při co nejvyšším výkonu)
- adaptivní systémy - systémy, v nichž se uplatňuje proces adaptivity, tj. vlastnost systému, která mu umožňuje reagovat na změny stavu systému a okolí tak, aby to bylo pro jeho existenci výhodné (např. zúžení zorniček při intenzívním světle)
- učící se systémy - systémy, které mění své vlastnosti (adaptují se) vlivem zpravidla opakovaných podnětů s cílem dosáhnout účelnějšího chování systému (např. systémy umělé inteligence)

10) *z hlediska oblasti zkoumání*

- systémy řízení - systémy s cílovým chováním, jejichž část působí na další systémy tak, aby dosáhly své žádoucí funkce
- systémy regulační - části systémů řízení, které pro svou činnost využívají zpětné vazby

- systémy informační - systémy, jejichž vazby jsou realizovány informacemi a prvky jsou místa transformace těchto informací; základem je účinná informační výměna mezi prvky systému a okolím, informační systém lze definovat jako soubor lidí, technických prostředků a metod zabezpečujících sběr, přenos, uchování a zpracování dat za účelem tvorby a prezentace informací.
- systémy komunikační - realizují přenos informací, jsou většinou tvořeny zdrojem zpráv, kodérem, přenosovým kanálem, dekodérem a příjemcem zprávy (telefonní okruhy)
- systémy interakční - informační systémy obyčejně s dálkovým přenosem a zpracováním dat, v nichž dochází k vzájemnému ovlivňování činnosti jednotlivých částí systému na základě vzájemného předávání a hodnocení informací (např. rezervace letenek, místenek, celostátní bankovní spojení apod.)
- systémy ekonomické - používané v oblasti ekonomického řízení a managementu, cílem je dosáhnout žádoucích vztahů mezi náklady a výnosy při respektování řady vedlejších cílů (např. řízení finančních prostředků, zásob, sledování trhu apod.)

11) *z hlediska charakteru dat, které zpracovávají*

- tvrdé systémy – představují systémy zpracovávající přesně strukturovaná data. Procesy probíhající v těchto systémech lze algoritmizovat, a tudíž mají jednoznačně zadané vstupy a výstupy. Příkladem mohou být průmyslové systémy, ve kterých člověk vystupuje jako tvůrce algoritmů a jako kontrolní element.
- měkké systémy – jsou systémy, které zpracovávají špatně strukturované a algoritmizované problémy. Tyto systémy pracují s celou řadou faktorů a vstupů, které nelze přesně kvantifikovat. Proto jsou tyto typy systému zatíženy velkou neurčitostí, riziky, nejistotami, případně nestabilitou. Důležitou roli v nich sehrává lidský faktor. Typickými „zástupci“ jsou firmy, banky, vzdělávací instituce, apod.

V reálném životě se nejčastěji vyskytují systémy kombinované, s některými problémy dobře s jinými obtížně strukturovanými a popsatelnými (identifikovatelnými).

Σ Shrnutí pojmů

Systém, prvek, vazba systému, okolí systému, vstup a výstup systému, subsystémy a dílčí systémy, struktura systému, hierarchie, hierarchická struktura, stav systému, chování a vlastnosti systému, operátor transformace systému, klasifikace systémů.

? Otázky

1. Charakterizujte pojmy systém, prvek a vazba systému.
2. Jaký je rozdíl mezi subsystémy a dílčími systémy?
3. Co znamená hierarchická struktura?
4. Čím se odlišuje proporcionální systém od integračního?

2. Modelování a simulace systémů

Čas ke studiu: 0.5 hodin

Cíl Po prostudování tohoto odstavce budete umět

- charakterizovat model jako nástroj pro zobrazení skutečnosti
- popsat proces simulace
- vysvětlit pojem identifikace systémů
- provést klasifikaci metod identifikace systémů

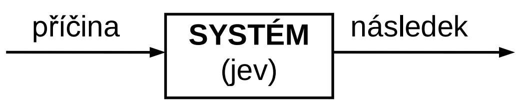
Výklad

Model

Často je nutné zkoumat chování různých zařízení v mezních situacích, které na skutečném zařízení nesmí nastat, neboť by mohly vést k značným škodám, nebo je nutno vyšetřit vlastnosti objektů ještě před jejich výrobou. V takových případech je velmi efektivní pracovat místo se skutečným zařízením s jeho modelem.

Model je pak zobrazení podstatných vlastností reálného (nebo konstruovaného) systému, které ve vhodné formě vyjadřuje informaci o systému. Musí vyjadřovat vztahy příčiny a následků. Příčina a následek jsou spolu prostřednictvím systému vázány operátorem transformace F_o.

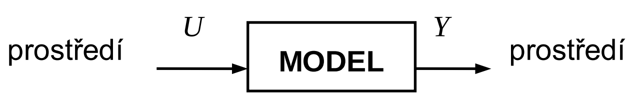
Schematicky můžeme tento vztah vyjádřit:



Obr. 6 Vztahy v modelu systému

Popis tohoto uspořádání budeme nazývat modelem. Přitom je jedno, pomocí jakého výrazového prostředku je tento popis proveden. Může být proveden matematicky, formou grafů, tabulek, algoritmem, ale také jen slovně.

Popis lze formalizovat:



Obr. 7 Formalizace popisu systému

Zde jsme označili:

- příčinu U (vstup modelu),
- následek Y (výstup modelu).

Vazbu mezi nimi lze zapsat ve tvaru

$$Y = F(U)$$

F je pravidlo, podle kterého přiřazujeme následek Y příčině U - přiřazujeme výstup modelu jeho vstupu. Toto pravidlo F nazýváme ***operátorem modelu***.

Model reálného systému je vždy spojen se zjednodušením a zanedbáním nepodstatných detailů reálného systému, protože reálná skutečnost může být lidským pozorovatelem vystižena jen do určité míry. Právě tato míra rozhoduje, jak přesně bude model vystihovat chování reálného objektu, ale současně určuje jeho složitost a tím i praktickou použitelnost (např. co nejpřesnější popis chování reálného systému by mohl vést k tak složitému modelu, že by pak nebyl prakticky použitelný). V této fázi tvorby modelu musíme rozlišit sledované jevy od nesledovaných, podstatné od nepodstatných

Matematický model je pak zobrazením podstatných vlastností reálného systému matematickým popisem.

Simulace systémů

Základní princip simulace systémů je nahrazení původního systému jiným systémem, tzv. ***simulačním modelem***, a zpětná aplikace poznatků ze simulačního modelu na původní systém.

Simulace představuje jeden z účinných nástrojů pro analýzu a racionalizaci řízení složitých procesů a systémů. Simulaci můžeme popsat jako proces tvorby modelu reálného systému včetně procesu experimentování s tímto modelem s cílem získat lepší informace o chování studovaného systému. Simulace pomáhá popsat a předvídat chování různých systémů a procesů z mnoha oborů lidské činnosti bez následků pro simulovaný objekt, jeho okolí, obsluhu, životní prostředí apod. Místo toho, abychom sledovali dynamické chování nějakého reálného sytému nebo procesu a jeho reakce na provedené změny, sledujeme chování jeho modelu. Můžeme sledovat chování systému za různých podmínek, v havarijních situacích, ve stavech, které např. na skutečném objektu nesmí nastat.

Rozvoj výpočetní techniky znamenal podstatné rozšíření možností řešitelnosti matematických modelů. Umožnil automatizovat výpočet relací matematického modelu. Technická realizace matematického modelu objektu na počítači se označuje jako ***počítačový model***. Počítačová realizace matematického modelu umožňuje automatizovat výpočet řešení rovnic modelu, čímž uživateli zůstává pouze zadávat vstupy matematického modelu a zpracovávat výstupy získané prostřednictvím výstupních zařízení počítače (grafické průběhy, tabulky). Při dostatečné programové podpoře můžeme s matematickým modelem systému experimentovat obdobně jako s reálným objektem.

Hlavní fáze procesu simulace:

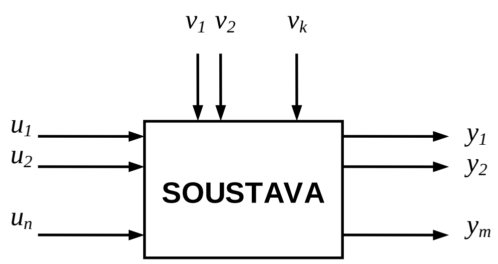
- vymezení systému na zkoumaném objektu, určení matematického popisu systému - *identifikace*
- sestavení modelu (realizace např. na PC) - *modelování*
- ověření shody (verifikace) projevů modelu a objektu - *simulace*

- vlastní experimenty s modelem - *simulace*
- aplikace výsledků simulačních experimentů na zkoumaný objekt

Identifikace systémů

Identifikace je proces určování matematického popisu modelu systému. Je to činnost, při které určujeme strukturu a parametry modelu. Je-li struktura známá, hovoříme o *odhadu parametrů*. Úloha identifikace spočívá v určení (syntéze) operátoru modelu *F*, tj. v provedení vyhodnocení měření a určení odhadu operátoru *F* tak, aby byl v určitém předem definovaném smyslu blízký skutečnému operátoru *Fo*.

Objekt identifikace si lze znázornit takto:



Obr. 8 Objekt identifikace

kde veličiny

$u_1, u_2, \ldots\ldots, u_n$	měřitelné vstupy soustavy,
$y_1, y_2, \ldots\ldots, y_m$	měřitelné výstupy soustavy,
$v_1, v_2, \ldots\ldots, v_k$	poruchové vstupy soustavy.

Zkoumaný systém lze identifikovat buď *analyticky*, tj. pomocí metod matematicko-fyzikální analýzy nebo *empiricky*, tj. pomocí metod experimentálních (deterministické a statistické metody). Praktické metody leží mezi těmito dvěma krajními případy. Vždy je užitečné teoretickou cestou najít přibližné matematické vztahy popisující daný systém a experimentálně pak upřesnit parametry, které v nich vystupují (tzv. deduktivní postupy).

1) Identifikace metodou matematicko-fyzikální analýzy

Identifikace metodou matematicko-fyzikální analýzy vychází ze známých přírodních zákonů, které umožňují popsat vztah mezi vstupní a výstupní veličinou soustavy. Při sestavování rovnic (nejčastěji soustava diferenciálních rovnic) vycházíme z hmotových nebo energetických bilancí z rovnic fyzikálních, chemických a biologických procesů (např. rovnice kontinuity, Fourierova rovnice sdílení tepla, sdílení tepla konvekcí, Bernoulliho rovnice).

Takovýto matematický model lze charakterizovat jako *„vnitřní" popis* chování zkoumaného systému. Takovýto model reprodukuje *skutečné zákonitosti*, jeho parametry mají fyzikální smysl, je většinou *složitý a nelineární*, ale popisuje chování systémů ve *„větším rozsahu"*. Vede k *jednoznačnému* popisu systému.

Výhodou této metody je to, že umožňuje určit matematický model v případech, kdy se soustava teprve projektuje. Výsledků takového rozboru lze užít pro volbu optimální koncepce a detailní konstrukce celého zařízení.

Nevýhodou je, že vyžadují důkladné teoretické znalosti příslušného oboru, kam identifikovaný objekt patří, a že výsledky jsou značně složité a výsledné vztahy nutno někdy aproximovat, linearizovat, a to na úkor přesnosti.

2) Experimentální metody identifikace

Při experimentálním způsobu identifikace (induktivní postupy) se matematický model určí na základě experimentálně obdržených údajů v chování daného systému. Předpokládá se přitom, že hodnoty vstupních a výstupních signálů lze měřit. Takovýto model lze charakterizovat jako *„vnější popis chování"* daného systému. Má většinou *jednoduchý tvar,* parametry se snadno určují, ale často nemají fyzikální smysl. Takový model je použitelný v *„menším rozsahu"* (např. v okolí pracovního bodu, v ustáleném stavu apod.). Charakter tohoto způsobu je *nejednoznačný*: pro systém můžeme získat několik popisů podle použité metody identifikace, zvolené struktury a složitosti modelu.

U experimentálních metod identifikace se předpokládá, že můžeme měřit přímo v provozu vstupní a výstupní signály systému a záznamů o časovém průběhu těchto signálů pak použít k vyhodnocení dynamických vlastností systémů. Rozborem vstupních a výstupních veličin systému můžeme získat matematický model vyjadřující jeho vnější popis (např. diferenciální rovnici, přenos apod.). Při experimentálních metodách předpokládáme aspoň přibližnou znalost struktury objektu, který považujeme za černou skříňku „black box". Nevýhodou těchto metod identifikace je skutečnost, že identifikovaný objekt je obvykle součástí většího zařízení, a proto jej nemůžeme zkoumat izolovaně, jak bychom si přáli. Při zkoumání v souvislosti s jiným zařízením se nám často nepodaří odstranit působení jiných veličin zařízení, ani poruchových veličin.

Deterministické metody experimentální identifikace neuvažují působení náhodných veličin na objektech ani nepřesnosti měření. Deterministické metody jsou jednoduché a názorné. Je-li měření na objektu provedeno pečlivě, dostaneme dobré výsledky. Hodí se především pro jednoparametrové soustavy. Pro víceparametrové soustavy se hodí tehdy, můžeme-li hodnoty nesledovaných veličin zanedbat nebo jejich vliv vyloučit (udržováním na konstantní hodnotě).

Statistické metody předpokládají působení náhodných veličin na objekt, nebo že měřené veličiny jsou zatíženy šumem. K identifikaci systémů podle náhodných časových průběhů vstupních a výstupních signálů slouží metody statistické dynamiky jako korelační analýza, regresní analýza a jiné. Statistické metody identifikace vyžadují mnohem větší soubory změřených dat a jejich zpracování je možné pouze s použitím počítače.

Při experimentálním způsobu identifikace postupujeme obvykle tak, že pro vhodně zvolenou *strukturu modelu* (tím rozumíme způsob matematického vyjádření závislosti výstupního signálu na signálu vstupním např. ve tvaru diferenciální rovnice, diferenční rovnice, přenosu, impulsní charakteristiky) provedeme odhad jeho *parametrů* tj. koeficientů rovnic, resp. přenosů. Provádíme to obvykle aplikací různých metod pro vyhodnocení záznamů odezvy systému na definovaný vstupní signál.

Výsledky experimentu lze však využít a zpracovat i jinými způsoby:

a) lze jich využít k praktickému ověření závěrů, vyplývajících z matematicko fyzikálního rozboru soustavy, případně k zpřesnění matematického modelu nalezeného cestou matematicko-fyzikální analýzy,

b) v některých případech umožňují identifikaci konstant vyjadřujících kvantitativně průběh procesu - jako jsou součinitelé přestupu tepla při ohřevu apod.

Naopak výsledků matematicko-fyzikální analýzy lze využít k odhadu řádu rovnice či přenosu identifikované soustavy při experimentální identifikaci.

Z uvedeného vyplývá, že obě metody se vhodně doplňují. Lze říci, že především vhodnou kombinací těchto metod je možno vytvořit předpoklady pro zajištění úspěchu identifikace. Nalezení nejvhodnějšího způsobu identifikace předpokládá velké teoretické znalosti a především určité zkušenosti, protože každý konkrétní systém vyžaduje jiný způsob identifikace.

Σ Shrnutí pojmů

Model, operátor modelu, matematický model, simulace systémů, identifikace systémů

? Otázky

5. Charakterizujte pojem model systému.
6. Popište proces simulace.
7. Vysvětlete pojem identifikace systémů.
8. Proveďte klasifikaci metod identifikace.

3. Řízení systémů

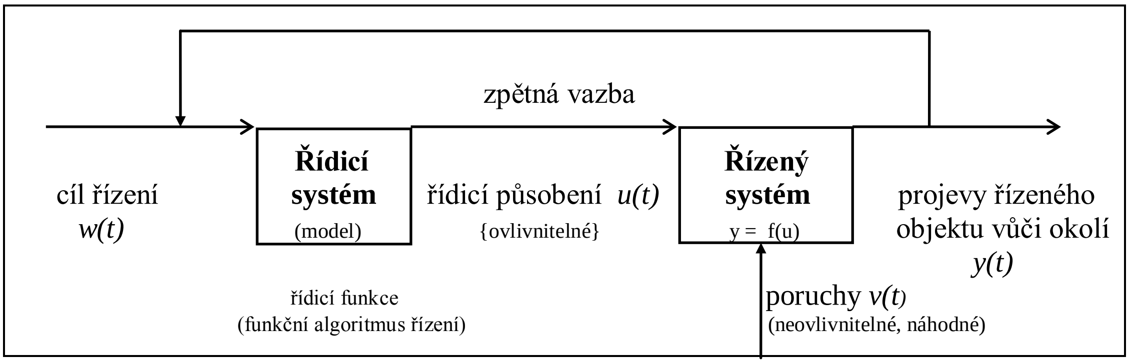
Čas ke studiu: 0.5 hodin

Cíl Po prostudování tohoto odstavce budete umět

- definovat pojem řízení systémů
- charakterizovat základní druhy řízení systémů
- popsat různé druhy regulace systémů

Výklad

Řízení chápeme jako cílevědomé působení řídicího systému na systém řízený za účelem dosažení vytýčeného cíle.



Obr. 9 Obecný princip řízení

Řídicí působení je prováděno podle určitého funkčního algoritmu, který je matematicky popsán tzv. řídicí funkcí, která je v podstatě matematickým vyjádřením vztahu mezi řídicím působením a cílem řízení (např. stavovou rovnicí, přenosem apod.)

Řídicí systém - fyzikální zařízení, které realizuje funkční algoritmus řízení tím, že generuje řídicí působení *u(t) na* řízený systém; matematickým popisem tohoto systému je tzv. řídicí funkce. Jako řídicí systém lze chápat např. člověka, regulátor, řídicí počítač apod.

Řízený systém - fyzikální zařízení, které chceme řídit (např. technologický proces, podnik), matematickým popisem abstrahujeme od jeho fyzikální podstaty a vytváříme model vlastního reálného objektu, který využíváme např. při simulaci systému na počítači.

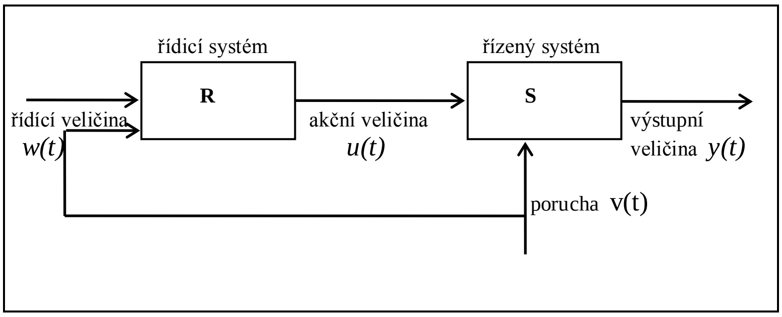
Rozlišujeme-li možné způsoby cílevědomého působení na systém, je účelné rozlišovat mezi *sledováním* (monitorováním systému) a vlastním *řízením* systému.

- *Sledováním (monitorováním)* systému rozumíme spojité či přetržité získávání informací o stavu systému bez současného působení na systém.
- *Řízení* je cílevědomé působení řídicího systému na systém řízený za účelem dosažení vytýčeného cíle

Rozlišujeme dva základní druhy řízení podle toho, zda je obvod řízení otevřen nebo uzavřen: *ovládání* a *regulaci*.

Ovládání (dopředné řízení = feedforward)

Jedná se o řízení při otevřeném obvodu. K řízení využívá jen apriorních informací o řízeném objektu a nijak se nekontroluje jeho skutečný stav (není zpětná vazba).

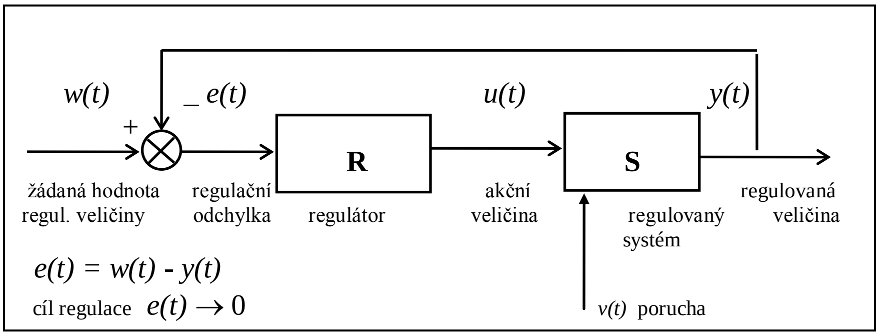


Obr. 10 Ovládání s měřením poruch

Ovládání je možno s úspěchem použít je tam, kde můžeme s jistotou tvrdit, že výstupní veličina řízeného systému *y(t)* bude přesně taková, jako ji předpokládá řídicí systém. Aby byl tento požadavek splněn, musí být do nejmenších podrobností znám matematický popis řízeného systému a podchyceny i všechny poruchy na tento systém působící. Opomenutí některých vazeb vede k nekontrolované odchylce skutečné hodnoty výstupní veličiny řízeného systému *y(t)* od požadované hodnoty *w(t)*. Proto se ovládání používá převážně u řízení logického (spínače, výtahy, semafory), kde vztah mezi výstupem a vstupem řízeného systému je popsán logickými funkcemi a výstup je svou povahou (logické 0 a 1) prakticky nezávislý na poruchách.

Regulace (řízení pomocí zpětné vazby = feedback)

Jedná se o řízení v uzavřeném obvodu. Zde se navíc k řízení využívá i informace o stavu řízeného systému, a to obvykle měřením výstupní veličiny řízeného systému *y(t)* (zpětná vazba).



Obr. 11 Regulace

Řídicí systém porovnává žádanou hodnotu regulované veličiny *w(t)* se skutečnou hodnotou regulované veličiny *y(t)*. Existuje-li mezi *w(t)* a *y(t)* odchylka *e(t)*, působí řídicí systém akční veličinou *u(t)* na řízený systém tak, aby byla odchylka odstraněna.

Cílem regulace je tedy udržení nulové (minimální) odchylky. Z popisu principu regulace je patrno, že se zde pro řízení bezprostředně nevyužívá matematického popisu řízeného systému a většinou se ani neměří poruchy vstupující do systému. Přesto je tento princip řízení tak univerzální, že dovoluje řídit systémy s nejrůznějšími dynamickými vlastnostmi, dokonce i některé systémy nestabilní. Matematický popis řízeného systému je využíván pro nastavení prvků řídicího systému, aby bylo dosaženo optimálního regulačního pochodu.

Druhy regulace:

1) ***stabilizace*** - regulace na konstantní žádanou hodnotu regulované veličiny *w(t)* = konst.
2) ***programová regulace*** - regulace, kdy se *w(t)* mění v čase podle předem stanoveného programu
3) ***vlečná regulace (kaskádní)*** - regulace, kdy se *w(t)* mění podle určité technologicky významné veličiny
4) ***extremální regulace*** - regulace, kdy se hledá extrém funkce dvou proměnných
5) ***optimální regulace*** - regulace, kdy se hledá optimum funkce většího množství proměnných (funkcí)
6) ***adaptivní regulace*** - regulace, kdy se v procesu řízení regulátor samostatně přizpůsobuje změnám regulované soustavy, používá se především tehdy, mění-li soustava v průběhu řízení nezanedbatelně své parametry (dynamiku, tj. odpory, kapacity)

Pro řízení složitých dynamických systémů se stále více používá kombinace obou typů řízení, kdy je řízení s využitím matematických modelů korigováno prostřednictvím zpětných vazeb - hovoříme o *kombinaci řízení feedforward a feedback.*

U vyšších stupňů hierarchické struktury ASŘ se setkáváme s pojmy řízení *off - line*, *on - line* a *in - line*.

Pojem ***off -line*** je používán v případě, že neexistuje přímé spojení procesu s počítačem, údaje do počítače nebo z počítače jsou předávány člověkem. Toto spojení se označuje jako počítač - rádce. Úloha počítače je zde redukována pouze na jeho využití jako prostředku pro automatizaci výpočtů, kdy umožňuje rychlé prošetření variant řízení a výběr nejlepší, ale do rozhodování přímo nezasahuje. Tento typ řízení je charakteristický pro nejvyšší stupně hierarchické struktury ASŘ (ASŘ podniku, ASŘ výroby).

Pojmy ***on -line*** a ***in -line*** souvisí s řízením v uzavřené smyčce. Počítač zde prostřednictvím jednotky styku s prostředím měří potřebné fyzikální veličiny z procesu a na základě ověřeného algoritmu řízení vypočítává nejvhodnější parametry daného procesu. V případě, že cílem počítače je koordinovat, kontrolovat a ovlivňovat automatické systémy řízení na úrovní jednotlivých procesů a to tak, že vypočítané hodnoty zadává jako řídící veličiny analogovým regulátorům hovoříme o řízení *on - line*. Pokud počítač bezprostředně řídí průběh procesu, a to tak, že jsou analogové regulátory nahrazeny programem počítače, hovoříme o řízení *in - line*, o *přímém číslicovém řízení* (*Direct Digital Control - DDC*). Tento typ řízení je charakteristický pro nejnižší stupně hierarchické struktury ASŘ (ASŘ TP).

Podle stupně automatizace, tj. podle účasti člověka na řízení rozlišujeme řízení:

1) ***automatické*** - realizované pouze technickými prostředky bez bezprostřední účasti člověka
2) ***automatizované*** - realizované technickými prostředky s částečnou bezprostřední účasti člověka na řízení
3) ***neautomatické (ruční řízení)*** - vlastní řídicí funkce jsou realizovány jen člověkem

Zvláštním druhem řízení je ***řízení v reálném čase***: řídicí systém pracuje tak, že doba odezvy řízeného systému na řídicí zásah je menší než doba přechodu systému z původního stavu do stavu nového.

Σ Shrnutí pojmů

Řízení systémů, obecný princip řízení, ovládání, regulace, druhy regulace, řízení off-line, on-line, in-line, řízení automatické, automatizované, neautomatické, řízení v reálném čase.

? Otázky

1) Charakterizujte pojem řízení systémů.
9. Jaké jsou základní druhy řízení systémů?
10. Popište různé druhy regulace systémů
11. Co znamená řízení off-line, on-line, in-line?

5. Statické systémy

Čas ke studiu: 2 hodiny

Cíl Po prostudování tohoto odstavce budete umět

- definovat pojem statický systém
- uvést příklady matematického popisu statických systémů
- vysvětlit princip kódování grafů a jejich číselné vyjádření pomocí matic
- vysvětit a popsat kritickou cestu v síťovém grafu

Výklad

Matematický popis statických systémů

Samotný pojem ***statický systém*** je abstrakce. U reálných systémů nelze prakticky oddělovat statickou stránku od dynamické, obě jsou spolu úzce spojeny, záleží pouze na schopnosti rozlišení. Jde tedy o to, že v určitém konečném časovém intervalu se nám určité vztahy, jevy, struktury jeví jako časově invariantní, neměnné. Členění systémů na statické a dynamické je tedy uměle zavedené z důvodů metodologických.

Obecný statický systém je relace R definována na kartézském součinu množin:

$X= X_1\, xX_2\, x...x\, X_n$

kde X_1 ... X_n jsou množiny, jejichž prvky nejsou časově proměnné ani funkcemi času.

Příkladem obecného statického systému je soustava m lineárních rovnic o n neznámých:

$a_{11}x_1 + a_{12}x_2 + \qquad + a_{1n}x_n = b_1$

$a_{21}x_1 + a_{22}x_2 + \qquad + a_{2n}x_n = b_2$

…

$a_{m1}x_1 + a_{m2}x_2 + \qquad + a_{mn}x_n = b_m$

Tato soustava nám definuje relace mezi formálními objekty - proměnnými x_i. Hodnoty formálních objektů tohoto systému X_i jsou hodnotami proměnnými x_i. Vlastnosti tohoto systému jsou vlastnosti, které vyšetřuje lineární algebra (konzistence, homogenita, určitost soustavy).

Obecnějším typem obecného statického systému může být soustava lineárních nerovností:

$a_{11}x_1 + a_{12}x_2 + \qquad + a_{1n}x_n + b_1 <=> 0$

…

$a_{m01}x_1 + a_{m2}x_2 + \qquad + a_{mn}x_n + b_m <=> 0$

kde <=> znamená některý ze znaků <, <, =, >, >.

Ještě obecnějším typem obecného statického systému může být soustava“

$f_1\ (x_1,x_2,\ldots,x_n) <=> 0$

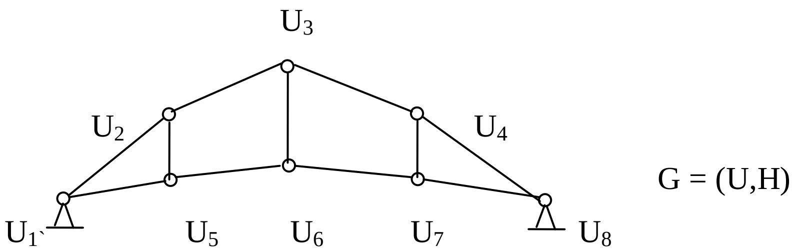
…

$f_m\ (x_1,x_2,\ldots,x_n) <=> 0$

Příklady statických systémů:

1) ***Prutová soustava***

Jde o zjednodušené znázornění kovového nosníku pomocí neorientovaného grafu, jež se používá ve statických výpočtech; v tomto případě zapisujeme styčníky jako uzly U ={U_1, U_2 .... U_8} a pruty jako hrany H = {H_1, H_2 .... H_{13}}.



Obr. 12 Prutová soustava

2) ***Leontiefské ekonomické modely***

Tyto modely popisují statické (strukturní) vztahy v ekonomických objektech pomocí orientovaných grafů. Prvky reálných systémů (uzly) jsou např. odvětví, vazbami (proudy) jsou toky zboží, výkonů a služeb, měřené zpravidla ve finančních jednotkách.

3) ***Modely soustav programů pro počítačové systémy***

Používají se v řízení technologických a ekonomických procesů. Analyzují se vzájemné vazby mezi programy v orientovaném grafu, který zobrazuje celou tuto soustavu. Uzly jsou programy a proudy jsou vzájemné vazby mezi těmito programy, mezi jejich vstupy a výstupy (výstupy získané z určitého programu se mohou používat jako vstupy do řady jiných programů).

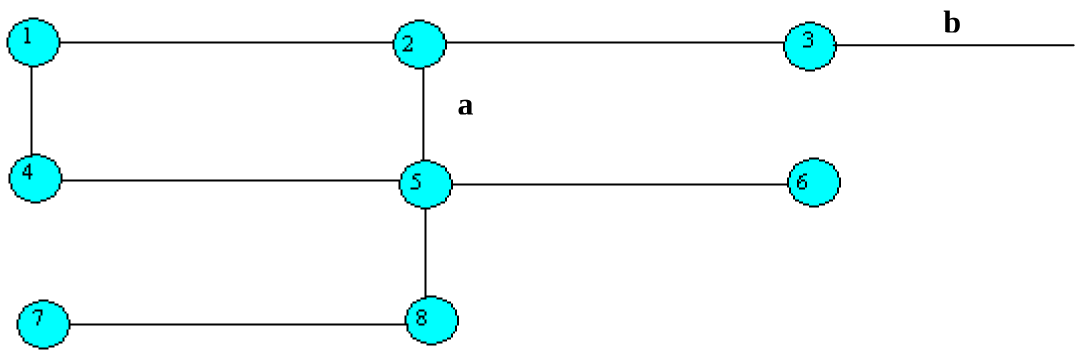
4) ***Modely automatizovaných systémů řízení***

V analýze a syntéze ASŘ se zobrazuje jejich struktura pomocí orientovaných grafů nebo matic Uzly grafu jsou činnosti (operace strojů, zpracování dat, určité rozhodování apod.) a vazbami mezi nimi jsou obvykle informace mezi činnostmi *i* a *j* . V matici pak binární hodnota prvku a_{ij} (0 nebo 1) označuje existenci či neexistenci informační vazby mezi činnostmi *i* a *j*.

Základní pojmy teorie grafů

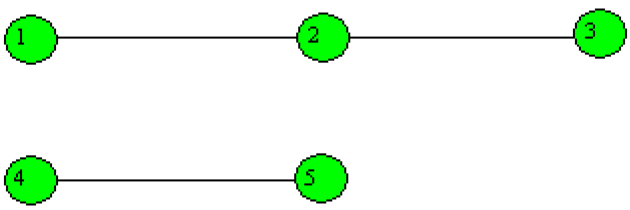
Teorie grafů je základem metod, jež využívá teorie systémů, operační analýza, strukturní analýza a syntéza. Její metody a postupy umožňují u systémů obsahujících velké množství prvků a vazeb mezi těmito prvky rychle se v nich orientovat a zjistit uvnitř systému i na rozhraní systému s okolím různé vztahy, které nás zajímají. Tyto metody a postupy se využívají především u ekonomických a výrobních procesů, kde je možno očekávat stovky až tisíce prvků. Jedná se např. o navrhování integrovaných obvodů počítačů, modelování technologických procesů, modelování vztahů v ekonomických systémech (toky zboží, výkonů a služeb).

Neorientovaný graf G(U,H) - je útvar obsahující *uzly* (U_1, ..., U_i, ..., U_I) a *hrany* (H_1, ..., H_j, ..., H_J). Každá hrana spojuje buď dva uzly, tzn.že je s nimi *koincidentní*, tehdy mluvíme o *vnitřní hraně* - ***a***, nebo je spojená s jedním uzlem útvaru a reprezentuje spojení s okolím - *vnější hrana* ***b*** (obr. 13).



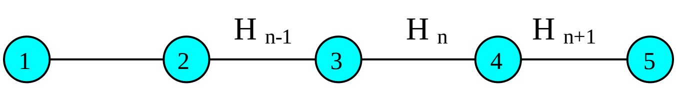
Obr. 13 Neorientovaný graf

Podgraf - podgrafem G' (U', H') grafu G (U, H) je graf, kde platí, že $U' \subset U$ a $H' \subset H$ (obr. 14).



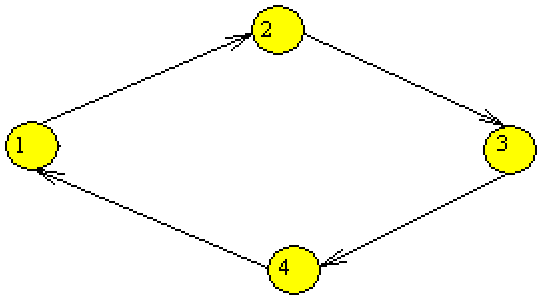
Obr. 14 Podgraf

Řetězec je pojem, pod kterým se rozumí *n* hran (H_1, . ., H_n,. ., H_N), které mají tu vlastnost, že jeden uzel koincidentní s libovolnou H_n (mimo H_1 a H_N) je spojený s H_{n-1} a druhý uzel koincidentní s H_n je spojený s H_{n+1}. Počet členů posloupnosti $\{H_n\}$ je zároveň *délkou řetězce* (obr. 15).



Obr. 15 Řetězec

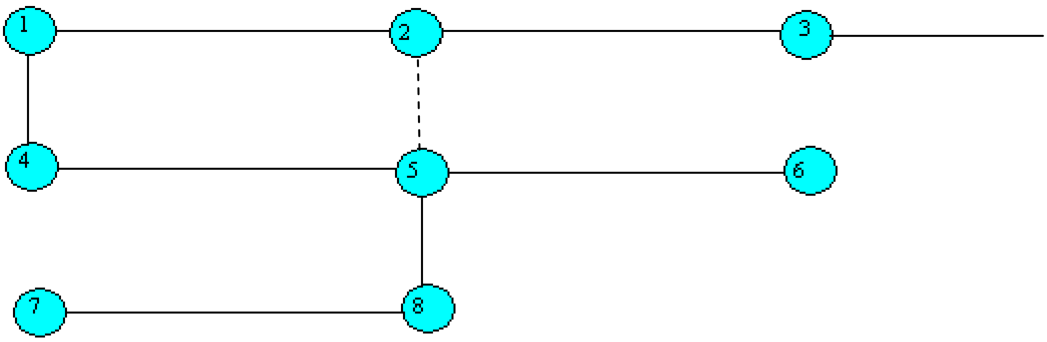
Cyklus - řetězec, který začíná a končí v jednom uzlu (obr. 16).



Obr. 16 Cyklus

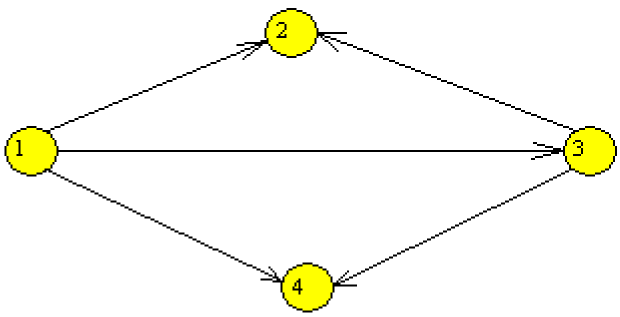
Spojený graf - graf, u kterého můžeme libovolné dva uzly spojit řetězcem (obr. 13, 15, 16, 17, 18).

Strom spojeného grafu - stromem nazýváme podgraf S (U, H'), který obsahuje všechny uzly grafu G, ale jen tolik hran, aby nevznikl cyklus. Hranám H' podgrafu říkáme *větve*, sestavující hrany grafu G nazýváme *tětivami* (obr. 17).



Obr. 17 Strom spojeného grafu

Orientovaný graf Q (U, P) - útvar obsahující nejen uzly, ale i orientované hrany - *proudy*, které mají všechny vlastnosti hran a navíc orientovanost toku (obr. 18).



Obr. 18 Orientovaný graf

Kódování grafů

Při kódování přiřadíme k jednotlivým uzlům indexy $k = 1, 2, \dots K$ a proudům $j = 1, 2, \dots. J$. Tento popis nám umožní jednoznačně číselně vyjádřit orientovaný graf pomocí *incidenční* a *asociační* matice (precedenční matice).

Incidenční matice I s rozměry *KxJ* definujeme následně:

pro prvek matice a_{kj} platí:

$a_{k,j}$= 1 - proud *j* je vstupním proudem do uzlu *k*

$a_{k,j}$= -1- proud *j* je výstupním proudem z uzlu *k*

$a_{k,j}$= 0 - proud *j* není koincidentní s uzlem *k*

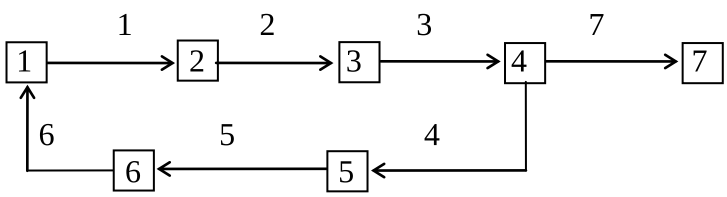
Při tvorbě ***precedenční (asociační) matice P*** se vychází z předpokladu, že libovolné 2 uzly jsou v jednom směru spojené nanejvýš jedním orientovaným proudem. V případě, že tento předpoklad není splněn, mohou se proudy buď sloučit, nebo lze graf doplnit fiktivními proudy nebo uzly. Výhodou precedenční matice je, že dostáváme matici dvouhodnotových funkcí „0“ a „1“. Precedenční matice P je čtvercovou maticí s rozměry *KxK*, pro jejíž prvky platí:

$p_{k,k'}$= 1 - za předpokladu, že existuje orientovaný proud z uzlu *k* do *k´*

$p_{k,k'}$= 0 - za předpokladu, že neexistuje orientovaný proud z uzlu *k* do *k´*

Řešený příklad 1

Vycházejme z obecného příkladu recyklu (obr. 19) a přiřaďme uzlům a proudům číselné hodnoty indexů *k* a *j*. Vyjádřete incidenční matici.



Obr. 19 Přiřazení indexů k prvkům orientovaného grafu

Řešení:

Incidenční matice bude mít tvar:

$$I = \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{vmatrix}$$

Precedenční matice bude mít tvar:

$$P = \begin{vmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

Precedenční matice nám poskytuje rychlou informaci o počtech vstupů a výstupů z jednotlivých prvků: ve sloupcích obsahuje přehled všech *předchůdců* (výstupy) jednotlivých prvků ve vzdálenosti jednoho proudu a v řádcích přehled všech *následovníků* (vstupy) těchto prvků ve vzdálenosti jednoho proudu (jedničkové prvky).

Můžeme např. určit největší počet vstupů, resp. výstupů daného prvku (najdeme největší součet jedniček ve sloupci, resp. v řádku), dále pak prvky bez výstupu, izolované prvky atd.

Asociační matice má významnou vlastnost: ***booleovské mocniny této matice P^n*** určují výskyt orientovaných řetězců v grafu. Je-li n počet proudů směřujících z uzlu k do $k´$, pak pro prvek matice P^n (n-té booleovské mocniny matice P) platí:

$p^n_{kk´}=1$ - pak z uzlu k do uzlu $k´$ směřuje orientovaný řetězec o délce n proudů

$p^n_{kk´}=0$ - neexistuje orientovaný řetězec o délce n proudů z uzlu k do uzlu $k´$

Matice P^n představuje seznam uzlů, které jsou od sebe vzdáleny o n hran (jedničkové prvky). *Předchůdci* daného prvku jsou prvky, které jsou počátečními uzly cest grafu, které v daném prvku končí.

Následovníci daného prvku jsou koncové uzly cest grafu systému, které v daném prvku začínají.

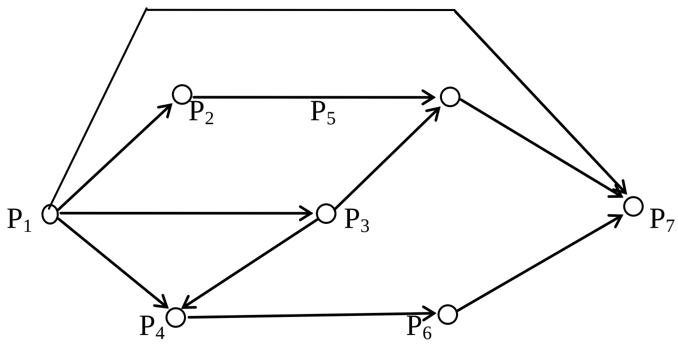
Jedničkové prvky $k´$-tého sloupce matice P^n nám udávají všechny předchůdce prvku $k´$o vzdálenosti dané exponentem n, tj. existenci *precedenčních cest* délky n k prvku $k´$.

Jedničkové prvky k-tého řádku matice P^n nám udávají všechny následovníky k-tého prvku ve vzdálenosti n, tj. všechny *sekvenční cesty* délky n k prvku k.

Matice P^n má velký význam v oblasti tvorby systémů detekce a korekce chyb např. v automatizovaných informačních systémech: zde nás zajímají všechna následující zpracování, do kterých může proniknout chyba vzniklá v určitém předchozím zpracování, nebo hledáme všechna předcházející zpracování, ze kterých se chyba mohla dostat do chybného zpracování.

Řešený příklad 2

Uvažujeme systém na obr. 20. Vyjádřete jej precedenční maticí a dále pak najděte všechny sekvenční a precedenční cesty délky n=2, 3, 4.



Obr. 20 Struktura systému znázorněného orientovaným grafem

Řešení:

$$P = \begin{vmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

$$P^2 = \begin{vmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

$$P^3 = \begin{vmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

$$P^4 = \begin{vmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

Z matice P^2 vidíme, že např. k prvku p_1 systému jsou prvky p_4 , p_5 , p_6 následovníky (řádky) ve vzdálenosti 2; k prvku p_2 jsou následovníci prvky p_6 a p_7 ve vzdálenosti 2. K prvku p_4 je následovník ve vzdálenosti 2 prvek p_7.

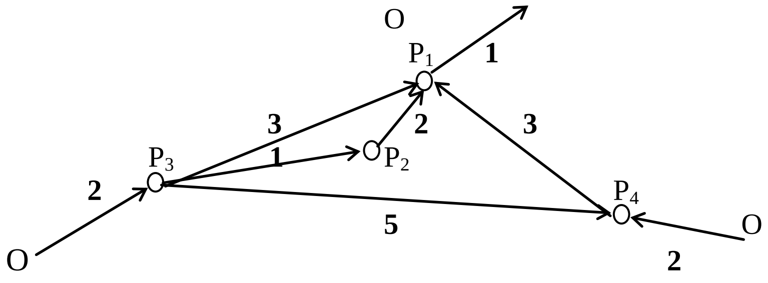
Čteme-li matici P^2 po sloupcích, pak prvek p_4 má předchůdce prvek p_1 ve vzdálenosti 2,

prvek p_5 má předchůdce prvek p_1 ve vzdálenosti 2, prvek p_6 má předchůdce prvky p_1 a p_3 ve vzdálenosti 2 a prvek p_7 má předchůdce prvky p_2, p_3 a p_4 ve vzdálenosti 2.

Pomocí dříve popsaných matic s booleovskými prvky, které mohou nabývat hodnot „0“ a „1“ je možno popisovat vazby pouze existenčně (říkají nám jen, zda existuje či neexistuje vazba mezi prvky k a k'). Tyto matice nám dávají málo informací o skutečné struktuře systému. Proto byly zavedeny *matice čísel*. Nahradíme-li dvouhodnotové prvky v precedenční matici čísly, dostáváme ***matici ohodnocení vazeb***. Těmito ohodnoceními mohou být různé parametry vazeb: *počet opakování vazeb*, *množství*, *vzdálenosti*, *doby trvání*.

Řešený příklad 3

Je dán systém znázorněný orientovaným grafem na obr. 21. Jednotlivé vazby jsou ohodnoceny parametry. *O* - označujeme okolí systému, které můžeme považovat za další prvek systému, tj. $O=p_5$. Určete matici ohodnocení vazeb.



Obr. 21 Struktura systému znázorněného orientovaným grafem

Řešení:

Matice ohodnocení vazeb má tvar:

$$A = \begin{vmatrix} 0 & 0 & 0 & 0 & 1 \\ 2 & 0 & 0 & 0 & 0 \\ 3 & 1 & 0 & 5 & 0 \\ 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 2 & 0 \end{vmatrix}$$

Úlohy o rozhraní (interface)

Analyzují se vlastnosti sousedních prvků, subsystémů, resp. prvků a vazeb, které do nich vstupují a vystupují. Vyšetřuje se konzistence parametrů na vstupech a výstupech sousedních prvků. V těchto úlohách se zkoumá, za jakých podmínek bude existovat shoda mezi přípustnými hodnotami parametrů na vstupech a výstupech sousedních prvků.

Řešený příklad 4

Uvažujeme již dříve popsaný systém na obr. 20, který může být zjednodušeným obrazem např. informačního nebo počítačového systému. Předpokládejme, že vstupy každého prvku systému jsou charakterizovány jedním skalárním parametrem $p_i^{vst.}$ a výstupy rovněž jedním skalárním parametrem $p_i^{výst.}$. Tyto parametry mohou charakterizovat např. druhy vstupů a výstupů, rytmus vstupů a výstupů (frekvenci), kódy vstupů a výstupů apod. Parametry vstupů a výstupů jsou zadávány tabulkou:

skalár. par. / prvky	p_j^{vst}	$p_i^{výst}$
p_1	2	1
p_2	1	2
p_3	2	2
p_4	1	1
p_5	3	1
p_6	2	3
p_7	1	3

Řešení úlohy spočívá v prověření shody parametrů na vstupech a výstupech sousedních prvků. Je-li u vazby v_{ij} splněna rovnost $p_i^{výst} = p_j^{vst} = p$, říkáme, že daná vazba v_{ij} je *regulární* a má parametr *p*. Není-li rovnost splněna, je vazba *neregulární*. Informace o existenci vazeb v_{ij} získáme z precedenční matice: jsou-li jedničky v průsečíku mezi prvky p_i a p_j, existuje vazba mezi těmito prvky. Srovnáním $p_i^{výst}$ a p_j^{vst} parametrů prověříme regulárnost vazby. Informaci o neregulárních vazbách zapisujeme do ***matice neregulárnosti vazeb R***: je-li vazba v_{ij} regulární, pak pro prvek matice $r_{ij} = 0$, není-li regulární, pak $r_{ij} = 1$.

Řešení:

Precedenční matice má tvar:

$$P = \begin{vmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

Matice neregulárnosti vazeb:

$$R = \begin{vmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix}$$

Matice R je řešením dané úlohy o rozhraní, používá se při úpravách rozhraní mezi prvky systému, kdy je nutno systém postupně upravovat tak, aby všechny vazby byly regulární (tj. aby matice R byla nulová). Úpravy se provádějí např. modifikací parametrů vstupů a výstupů, záměnou prvků, modifikací celkové struktury systému apod.

Metoda kritické cesty (Critical Path Method)

Kritická cesta - křivka, která spojuje počáteční a koncový uzel síťového grafu přes tzv. kritické činnosti, tj. takové, jejichž celková časová rezerva je nulová, jejichž zdržení by způsobilo prodloužení celkového času pro realizaci určitého projektu. Probíhá celým grafem a vždy je nejméně jedna.

Cíl: zajistit co nejdříve možný konec akce.

Použití: plánování a realizace akcí (např. plánování a realizace ekonomických i technologických procesů, plánování a realizace různých oprav, staveb apod.) Při řešení realizačních prací dává možnost předem určit nároky např. na časové zatížení pracovníků, spotřebu materiálu, simulovat dopředu vznik možných situací a tím se na ně připravit.

Pro řešení metodou kritické cesty využíváme tzv. *síťový graf*, který se skládá z uzlů a orientovaných hran. Hrany odpovídají jednotlivým dílčím činnostem úkolu. Danou činnost jednoznačně určují počáteční a koncový uzel, kterými je každá činnost ohraničena. Činnost označujeme uspořádanou dvojici čísel (i, j), přičemž musí platit $i < j$.

Na realizaci činnosti je třeba určité doby, tzv. doby trvání činnosti t_{ij}, a vynaložení určitých nákladů. Některé činnosti musí být vykonány v určitém časovém pořadí, proto je třeba do síťového grafu zavést ještě fiktivní činnosti s nulovou dobou trvání, které vyjadřují vazby a závislosti mezi činnostmi. Síťový graf má tyto základní vlastnosti:

• Každý síťový graf musí mít vždy jeden počátek, ze kterého hrany pouze vystupují, a jeden konec, do kterého hrany pouze vstupují. Tuto podmínku lze splnit vždy pomocí fiktivních činností (obr. 22 a).

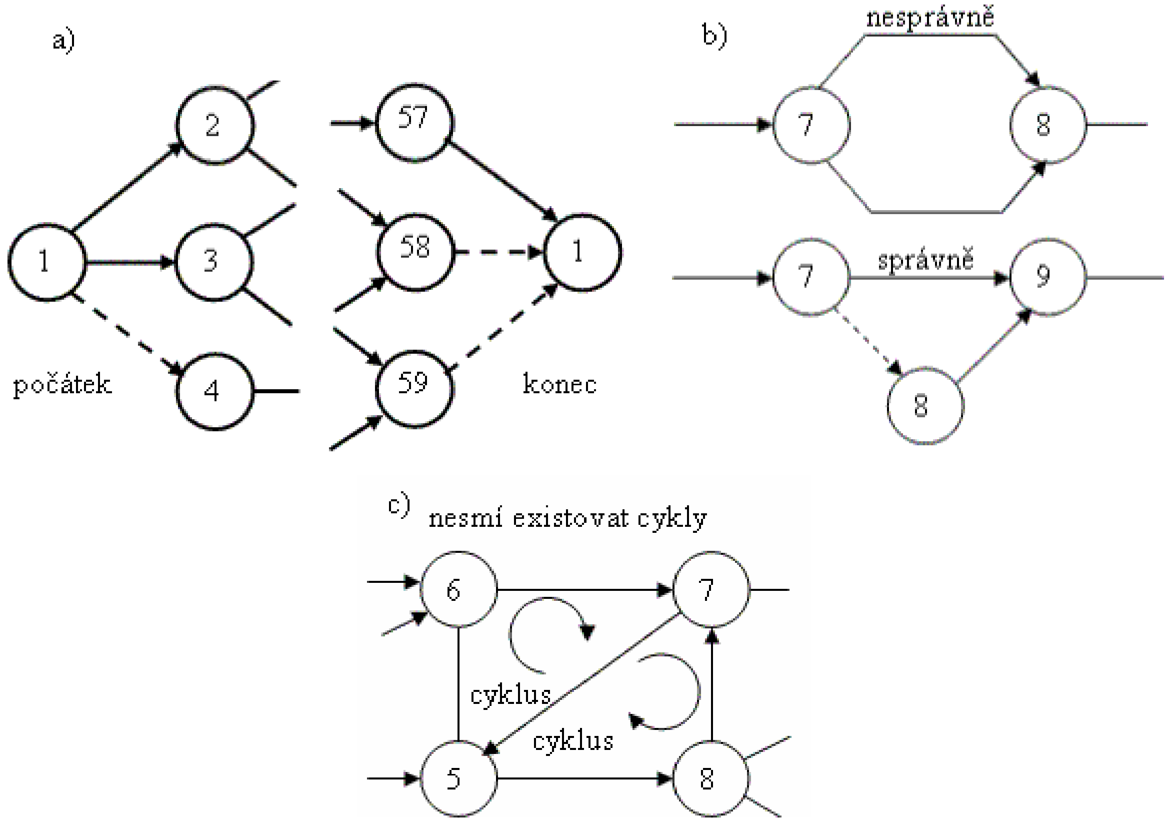
• Každá činnost může být zahájena jen tehdy, když jsou dokončeny všechny předcházející činnosti.

• Souběžné (paralelní) činnosti z důvodu jednoznačné identifikace musí být odděleny fiktivní činností (obr. 22 b).

• Délky hran neodpovídají dobám trvání činností.

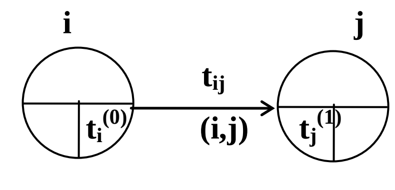
• Uzly lze očíslovat tak, aby platila nerovnost *i* < *j*. V tomto případě v síťovém grafu nevystupují cykly (obr. 22 c).

Při sestavování síťových grafů je třeba provést rozbor činností a uvědomit si, které činnost bezprostředně předcházejí každé činnosti, které činnosti za danou činností bezprostředně následují, které činnosti probíhají souběžně a které činnosti na sobě nezávisejí. Všechny údaje o činnostech zapisujeme do tabulky, na základě které pak sestavíme vlastní síťový graf.



Obr. 22 Základní vlastnosti síťových grafů pro použití metody CPM

<u>Symbolika CPM</u>



Obr. 19 Popis uzlů a hran v síťovém grafu

2) $\mathbf{t_i}$ **,** $\mathbf{t_j}$ - *termín splnění* - čas, kdy má být určitá etapa dokončena v uzlu i ,j

3) **i , j** - *pořadí uzlů* - znázorňujeme v horní části kruhu

4) **(i,j)** - *činnost* - graficky vyjádřena jako úsečku, která vychází z uzlu i do j , směr průběhu činnosti je označen šipkou

5) $\mathbf{t_{ij}}$ - *doba trvání příslušné činnosti* - zapisujeme do středu úsečky; aby byl projekt splněn v termínu musí platit pro úsek mezi uzly i a j : $\mathbf{t_i + t_{ij} \leq t_j}$

6) $\mathbf{t_i^{(0)}}$ - *nejdříve možný začátek činnosti (i,j)* - určuje čas, kdy jsou splněny všechny činnosti, které je nutno provést do uzlu i, a tedy může být zahájena činnost (i,j); je tedy dán trváním nejdelší cesty, která vede v síťovém diagramu do uzlu i; pro uzel 0 platí obyčejně $\mathbf{t_i^{(0)} = 0}$.

7) $\mathbf{t_i^{(1)}}$ - *nejpozději nutný začátek činnosti (i,j)* - vystihuje skutečnost, že v rozvětvených diagramech, kdy nejdříve možný začátek je dán trváním nejdelší cesty vedoucí do uzlu i, vznikají na ostatních cestách časové rezervy. Znamená to prakticky, že některé činnosti mohou být zahájeny později, aniž by došlo ke zpoždění hlavního cíle projektu a prodloužení celkové průběžné doby daného projektu.

8) $\mathbf{t_j^{(0)}}$ - *nejdříve možné ukončení činnosti (i,j)* - je vypočítáván za uvažování nejdříve možného začátku: $\mathbf{t_j^{(0)} = t_i^{(0)} + t_{ij}}$

9) $\mathbf{t_j^{(1)}}$ - *nejpozději nutné ukončení činnosti (i,j)* - určuje termín, kdy musí být splněna činnost (i,j) tak, aby bylo dosaženo konečného termínu celkového trvání projektu:

$$\mathbf{t_j^{(1)} = t_i^{(1)} + t_{ij}}$$

Postup při určování kritické cesty:

1) Nakreslíme síťový diagram, vnitřek uzlu rozdělíme na 3 části: v horní části číslujeme pořadí uzlů, do středů úseček zaznamenáváme dobu trvání příslušné činnosti $\mathbf{t_{ij}}$.

2) Výpočet hodnot $\mathbf{t_i^{(0)}}$ a $\mathbf{t_j^{(0)}}$ (nejdříve možného začátku a konce činnosti (i,j)):

   a) Jde-li o začátek procesu, volíme $t_i^{(0)} = 0$ (v případě, že řešíme dílčí proces, pak může být $t_i^{(0)} > 0$). Hodnotu $t_i^{(0)}$ vpisujeme do levé části uzlu i.

   b) K hodnotě $t_i^{(0)}$ připočítáme dobu t_{ij} (zapsanou ve středu úsečky) a výsledek $t_j^{(0)}$ píšeme na pravou část úsečky (k šipce). Postupujeme tak ve všech směrech z uzlu i.

   c) Končí-li v uzlu j jedna nebo více činností, pak maximální hodnotu $t_j^{(0)}$ ze všech vpisujeme do levé části uzlu, přičemž maximální hodnotu $t_j^{(0)}$ výrazně označíme orámováním.

3) Výpočet hodnot $\mathbf{t_i^{(1)}}$ a $\mathbf{t_j^{(1)}}$ nejpozději nutného začátku a konce činnosti (i,j):

   a) Při výpočtu vycházíme z konečného uzlu a postupujeme opačným směrem k počátečnímu uzlu. Hodnotu $t_j^{(1)} = t_j^{(0)}$ vpisujeme do pravé části uzlu j

   b) Od hodnoty $t_j^{(1)}$ odečítáme t_{ij} a výsledek píšeme na levou část úsečky jako $t_i^{(1)}$.

   c) Začíná-li v uzlu i jedna nebo více činností, pak minimální hodnotu $t_i^{(1)}$ ze všech vpisujeme do pravé části uzlu i, přičemž minimální hodnotu $t_i^{(1)}$ výrazně označíme orámováním.

Časová rezerva - doba, o kterou může být činnost zdržena, aniž by se ovlivnila celková doba trvání celé akce.

Kritická cesta v diagramu je určena sledem kritických uzlů, u nichž hodnoty uvedené v levé a pravé části se sobě rovnají, tj. nevykazuje žádné časové rezervy.

Řešený příklad 5

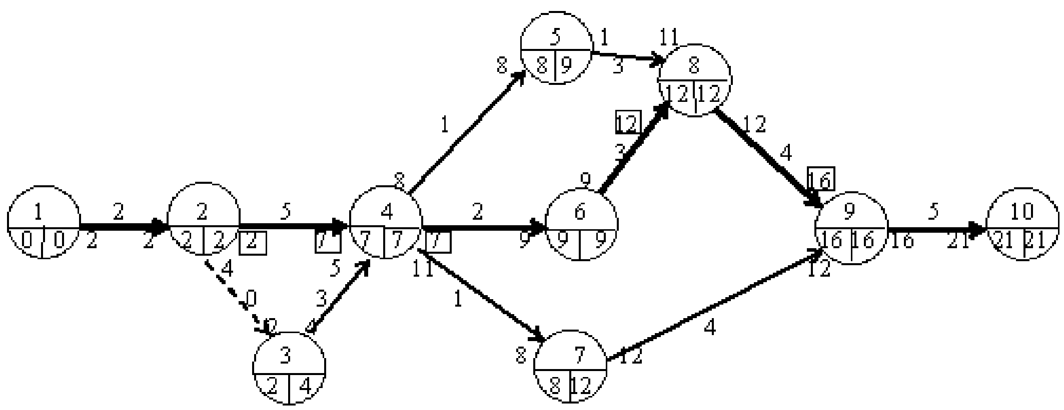
Máme za úkol sestrojit výrobek, který je vyroben ze tří součástí a tento příklad zakreslit do síťového grafu. Činnosti jsou označeny písmeny velké abecedy. Například symbol *A* < *B*, *C* označuje, že činnost *A* předchází činnostem *B* a *C* a tak dále. Podle tabulky 1 sestrojte síťový graf a určete kritickou cestu a časové rezervy. Doby trvání činností jsou uvedeny ve zvolených časových jednotkách (č.j.).

Tabulka 1 Vypočtené hodnoty k příkladu 2

	činnost	i	j	t_{ij}	podmínky
A	vstupní návrh	1	2	2	*A* < *B*, *C*
B	fiktivní činnost	2	3	0	*C* < *E*, *F*, *G*
C	rozbor	2	4	5	*B* < *E*, *F*, *G*
D	projekt	3	4	3	
E	objednávka 1	4	5	1	*E* < *H*
F	objednávka 2	4	6	2	*F* < *I*
G	objednávka 3	4	7	1	*G* < *J*
H	dodávka součástky 1	5	8	3	*H* < *K*
I	dodávka součástky 2	6	8	3	*I* < *K*
J	dodávka součástky 3	7	9	4	*J* < *L*
K	dílčí montáž	8	9	4	*K* < *L*
L	konečná montáž	9	10	5	

Řešení:

Nejdříve sestrojíme síťový graf s uvažováním omezení na časové činnosti. Přečíslování uzlů nemusíme provádět, protože vyhovuje podmínce i < j.



Obr. 23 Síťový graf

Výpočet nejdříve možných a nejpozději přípustných termínů provedeme přímo v síťovém grafu.

Kritická cesta: 1 – 2 – 4 – 6 – 8 – 9 – 10 (zesílenou čarou).

Doba realizace celého úkolu: $T\ 10 = 21$ č. j.

Σ Shrnutí pojmů

Matematický popis a příklady statických systémů, teorie grafů, incidenční matice, precedenční matice, sekvenční a precedenční cesty, matice ohodnocení vazeb, matice neregulárnosti vazeb.

? Otázky

1. Jakým způsobem matematicky popisujeme statické systémy?
2. Uveďte příklady statických systémů?
3. Popište matice, pomocí kterých lze číselně vyjádřit orientovaný graf.
4. Co vyjadřuje matice neregulárnosti vazeb?
5. Co je to kritická cesta v síťovém grafu?

6. Dynamické systémy

Čas ke studiu: 5 hodin

Cíl Po prostudování tohoto odstavce budete umět

- definovat pojem dynamický systém
- uvést příklady matematického popisu dynamických systémů
- definovat základní vlastnosti Laplaceovy transformace
- vysvětlit popis systému přenosem v Laplaceově transformaci
- provést klasifikaci základních lineárních dynamických soustav

Výklad

Dynamické systémy jsou systémy, jejichž stav se s časem mění, jsou závislé na čase.

Formální definice dynamického systému: dynamický systém je relace definovaná na kartézském součinu množin:

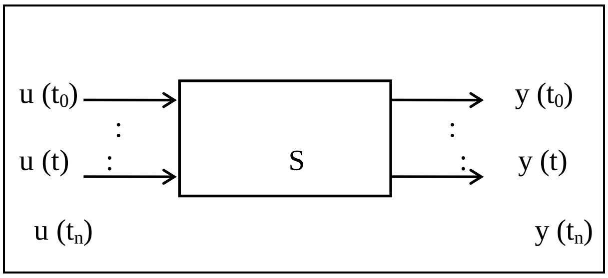
$S = S1 \times S2 \times \ldots \times Sn$

kde *S1* ... *Sn* jsou množiny, jejichž prvky jsou funkcemi času.

Obecně nepředpokládáme, že okamžitý výstup reálného systému závisí pouze na okamžitém vstupu. Výstup $y(t)$ obvykle závisí nejen na současném vstupu $u(t)$, ale i na minulé historii systému : tyto údaje o minulém vývoji systému vkládáme obvykle do stavového vektoru $z(t)$. Na stav systému se pak díváme jako na určitou paměť minulých příčin (vstupů).

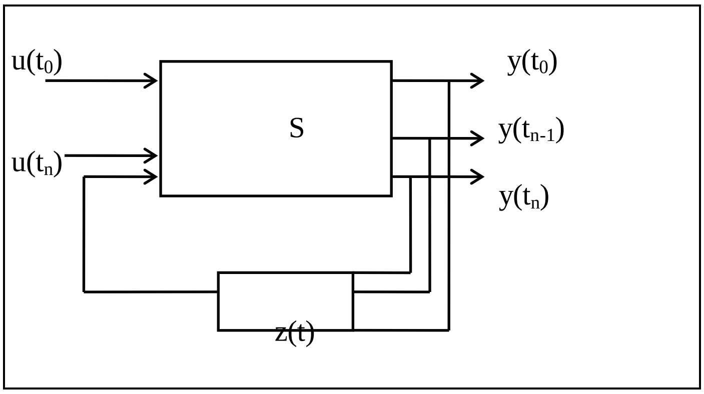
Z tohoto hlediska pak dynamické systémy dělíme na:

1. ***systémy typu vstup - výstup (input - output)*** - takové systémy, u kterých soustava množin *S1* ... *Sn* obsahuje pouze 2 kategorie množin, a to vstupy $u(t)$ a výstupy $y(t)$ (např. kombinační logické obvody, kdy výstup je kombinací vstupních proměnných).



Obr. 24 Dynamický systém typu vstup-výstup

2. ***systémy stavové*** - soustava množin *S1* , ... , *Sn* obsahuje alespoň jednu množinu stavů $z(t)$ (např. sekvenční logické obvody, kdy výstupní funkce je kombinací vstupních proměnných a stavu výstupní funkce).



Obr. 25 Dynamický systém stavový

Matematický popis dynamických systémů

Při matematickém popisu reálných zařízení zpravidla obdržíme nelineární matematické vztahy, které značně komplikují další využití výsledků. Zabýváme se proto rozborem, zda nelinearita je odstranitelná, tj. zda systém lze popsat lineárním matematickým modelem a pro řešení použít efektivních metod teorie lineárních regulačních obvodů anebo zda nelinearita je neodstranitelná a potom je nutné pro řešení použít metod teorie nelineárních regulačních obvodů.

Protože s nelineárními modely se pracuje obtížněji než s lineárními, snažíme se tyto nelineární modely nahradit modely lineárními, tj. nelineární modely linearizovat vytvořením totálního diferenciálu. Můžeme linearizovat jak statické, tak i dynamické vlastnosti daného systému. Musíme však zavést a potom i dodržet určité předpoklady, nejčastěji vymezením pracovní oblasti v blízkosti okolí pracovního bodu systému. Pro malé změny vstupních a výstupních signálů můžeme předpokládat, že vztah mezi nimi je lineární, tj. vyjádřitelný lineárními diferenciálními rovnicemi.
Je několik metod linearizace matematického popisu soustav (např. rozvoj nelineárního vztahu v řadu a využití pouze lineárních členů; linearizace popisu prvků, z kterých se soustava skládá; linearizace pomocí aproximace metodou nejmenších čtverců apod.).

Vyjádření matematického popisu spojitých lineárních soustav

Tato kapitola uvádí nejdůležitější tvary lineárních matematických modelů a třídění základních dynamických členů. Jsou zde uvažovány spojité lineární stacionární dynamické soustavy, tj. takové, jejichž vlastnosti se v čase nemění. Jejich matematické modely mohou mít různé tvary a mohou popisovat jejich vlastnosti v různých oblastech (prostorech). Proto je třeba vždy vybrat takový matematický model, který pro daný účel bude nejvhodnější. V teorii lineární regulace se nejčastěji pracuje s matematickými modely v níže uvedených tvarech, ze kterých lze určit všechny důležité dynamické i statické vlastnosti uvažovaných lineárních soustav.

Jednou z nejdůležitějších vlastností je fyzikální realizovatelnost, která spočívá v tom, že daný matematický model může popisovat (reprezentovat) nějaký skutečný lineární dynamický člen. Podmínka fyzikální realizovatelnosti v podstatě vyjadřuje princip příčinnosti (kauzality), který znamená, že následek (reakce, výstup) je vyvolán vždy příčinou (akcí, vstupem). Prakticky to znamená, že odezva dynamického členu nemůže vzniknout před okamžikem změny vstupní veličiny.

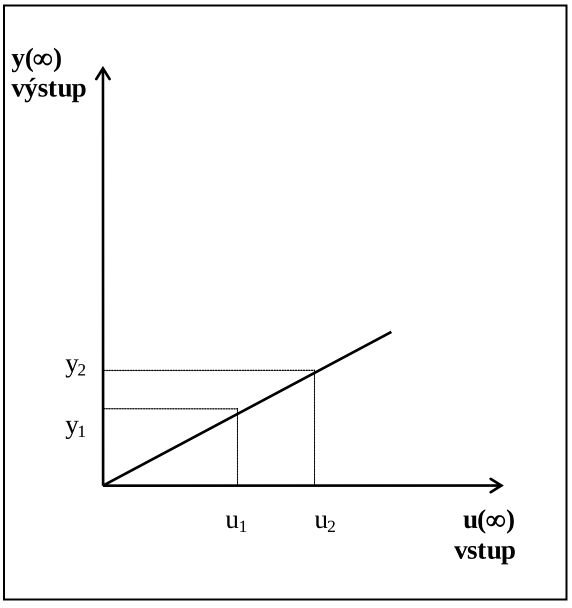
Základní tvary matematických modelů lineárních stacionárních dynamických soustav (se soustředěnými parametry) jsou:

- **lineární diferenciální rovnice**
- **přenos v Laplaceově transformaci**
- **přechodová funkce (charakteristika)**
- **impulsní funkce (charakteristika)**
- **frekvenční funkce (charakteristika)**
- **frekvenčním přenos**
- **odezva systému na libovolný vstupní signál**
- **stavové rovnice.**

Vlastnosti jakéhokoliv systému můžeme pozorovat za různých podmínek:

- v ustáleném stavu - pak hovoříme o ***statických vlastnostech***
- ve stavu neustáleném - pak hovoříme o ***dynamických vlastnostech***

Statické vlastnosti soustavy v ustáleném stavu se dají vyjádřit ***statickou charakteristikou*****,** která vyjadřuje závislost ustálených hodnot výstupní veličiny na ustálených hodnotách vstupní veličiny. U lineárních dynamických členů statická charakteristika, pokud existuje, je vždy přímka procházející počátkem souřadnic. Obecně však bývá tato závislost nelineární, pak provádíme její linearizaci v okolí pracovního bodu již dříve uvedenými metodami.



Obr. 26 Statická charakteristika

Nejjednodušším způsobem zjišťování této závislosti u dynamických soustav, je metoda určování bod po bodu. Postupně nastavujeme hodnoty vstupní veličiny a měříme odpovídající hodnoty výstupní veličiny. Přitom dbáme, aby se neuplatnila dynamika systému. To zajistíme odečítáním výstupní veličiny až po určité době od změny vstupní veličiny, tj. až se soustava dostane do ustáleného stavu.

V lineární diferenciální rovnici (viz. dále) vyjadřuje tuto závislost poslední člen rovnice, protože v ustáleném stavu vymizí všechny změny v čase (derivace):

$a_0\, y(t) = b_0\, u(t)$

pro $t \to \infty$ (takto vyjadřujeme ustálený stav, pro různé systémy je čas potřebný k dosažení ustáleného stavu různý: 1 min, 1 h , atd.)

$$k = \frac{b_0}{a_0} = \frac{y(\infty)}{u(\infty)}$$

Člen a_0, který popisuje statické vlastnosti soustavy. Statickou charakteristikou pak bude přímka:

$y(\infty) = k\, u(\infty)$

Směrnice přímky bude:

$$tg\ \alpha = k = \frac{b_0}{a_0}$$

Na základě rovnice statické charakteristiky můžeme dopředu určit, jaký bude výstup při daném vstupu v ustáleném stavu, ale rovnice nám nic neříká o tom, za jakou dobu tohoto výstupu dosáhneme. Nezískáme ještě úplný popis systému, nemůžeme jej ještě řídit.

Pro úplný popis systému musíme zjistit také ***dynamické vlastnosti*** systému, tedy závislost výstupní veličiny na čase. Pro určení dynamických vlastností soustav zavádíme na vstup soustavy předem definované signály. Tyto signály jsou buď *neperiodické*, nebo *periodické*.

Mezi nejčastěji používané *neperiodické* vstupní signály patří:

- Heavisideův jednotkový skok $\eta(t)$,
- Diracův jednotkový impuls $\delta(t)$,
- jednotkový skok rychlosti $\xi(t)$ resp. $v(t)$,
- jednotkový skok zrychlení $a(t)$ (užívaný méně často)

Mezi nejčastěji používané *periodické* signály patří:

- sinusový průběh
- sled pravoúhlých impulsů
- sled lichoběžníkových impulsů
- sled trojúhelníkových impulsů

Heavisideův jednotkový skok $\eta(t)$

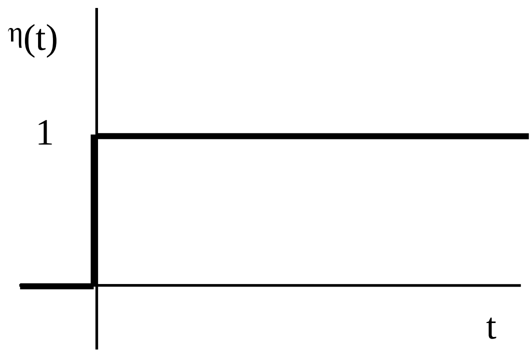
Je definován vztahy:

$\eta(t) = 1$ pro $t \geq 0$

$\eta(t) = 0$ pro $t < 0$

Laplaceovým obrazem $\eta(t)$ je:

Odezva na jednotkový Heavisideův skok je ***přechodová funkce h(t).*** Jejím grafickým vyjádřením je ***přechodová charakteristika***.



Obr. 27 Heavisideův jednotkový skok

Diracův jednotkový impuls $\delta(t)$

Představujme si, že vzniká z impulsu o výšce *h* a šířce *b*. Plochu impulsu si zvolíme jednotkovou a při současném zmenšování šířky ($b \to 0$) zvětšujeme výšku ($h \to \infty$) tak, aby stále platilo:

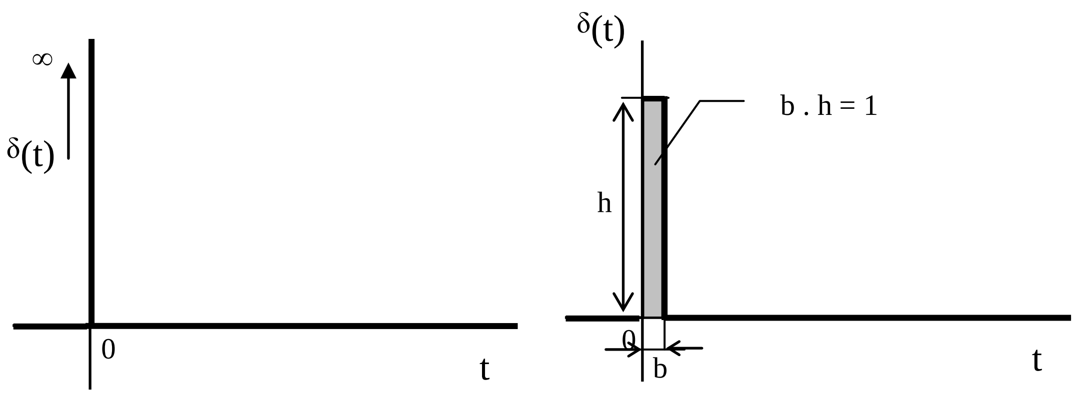
$$b \cdot h = 1$$

Diracův impuls je idealizovaná funkce fyzikálně nerealizovatelná. Lze ji charakterizovat vztahy:

$\delta(t) = 0$ pro $t = 0$

$\delta(t) = \infty$ pro $t \neq 0$

$$\int_{-\infty}^{\infty} \delta(t)dt = 1$$



Obr. 27 Diracův impuls

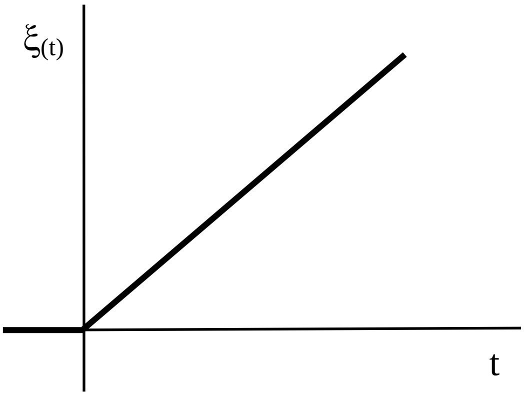
Odezvou systému na vstupní signál ve tvaru Diracova impulsu je ***impulsní funkce g(t)***. Jejím grafickým vyjádřením je ***impulsní charakteristika***.

Skok rychlosti $\xi(t)$

Je definován vztahy:

$\xi(t) = 0$ pro $t < 0$

$\xi(t) = t$ pro $t \geq 0$



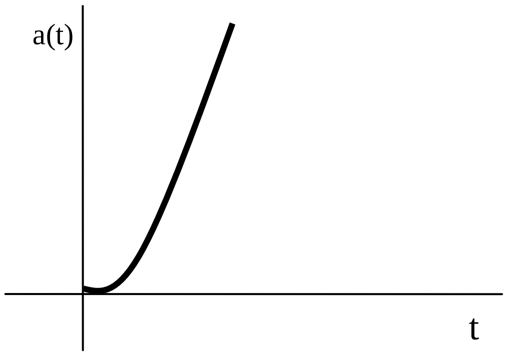
Obr. 28 Skok rychlosti

Skok zrychlení a(t)

Je definován vztahy:

$a(t) = 0$ pro $t < 0$

$a(t) = \frac{1}{2} t^2$ pro $t \geq 0$



Obr. 29 Skok zrychlení

Vstupní funkce ve tvaru skoku rychlostí, jednotkového skoku a jednotkového impulsu si navzájem odpovídají podle vztahů:

$$\delta(t) = \eta'(t) = \xi''(t)$$

$$\eta(t) = \xi'(t)$$

Diracův impuls je derivací jednotkového skoku a druhou derivací skoku rychlosti; jednotkový skok je derivací skoku rychlosti.

Podobně lze hovořit i o vztahu mezi odezvami. Impulsní charakteristika *g(t)* je derivací přechodové charakteristiky *h(t)*:

$$g(t) = h'(t)$$

Podobně je přechodová charakteristika derivací odezvy na skok rychlosti. Při praktické realizaci je nutno počítat s tím, že průběhy vstupních signálů nebudou ideální, ale dojde k jejich zkreslení.

Popis lineární diferenciální rovnicí

Jednou ze základních metod je popis systému lineární diferenciální rovnicí.

Obecný tvar popisu je

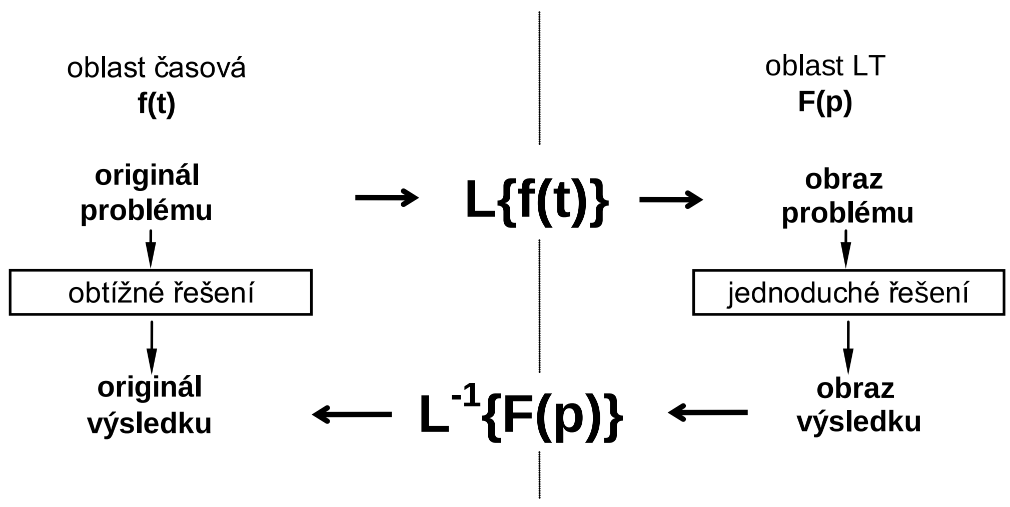
$$a_n y^{(n)}(t) + a_{n-1} y^{(n-1)}(t) + \dots + a_0 y(t) = b_m u^{(m)}(t) + \dots + b_0 u(t)$$

kde a_i , b_j jsou konstantní koeficienty, *u (t) je* vstup, *y (t) je* výstup systému. Z podmínky fyzikální realizovatelnosti plyne $m \leq n$. Řád diferenciální rovnice je roven řádu systému. Řešení rovnice je možno získat, máme-li určeny počáteční podmínky $y^{(n-1)}(0), \dots, y(0)$ a $u^{(m-1)}(0), \dots, u(0)$ a tvar vstupního signálu *u(t)*.

Popis přenosem v Laplaceově transformaci

Často v regulační technice, tedy i v popisu přenosových členů, užíváme tzv. Laplaceovy transformace (dále LT). LT patří do oblasti tzv. symbolických počtů, kde se dané funkci při výpočtech přiřazuje jiná funkce, která tuto původní funkci při výpočtech zastupuje a umožňuje podstatné zjednodušení prováděných matematických operací.

Je to integrální transformace, která převádí matematické operace jako je derivace nebo integrace v časové oblasti na násobení nebo dělení operátorem transformace *p*. Použitím této transformace lze některé obtížně řešitelné úlohy v časové oblasti převést na jednoduché řešení v operátorové oblasti podle schématu znázorněného na obrázku, kde je symbolem $L\{f(t)\}$ označena přímá transformace funkce času, symbolem $L^{-1}\{F(p)\}$ pak zpětná transformace Laplaceova obrazu do časové oblasti.



Obr. 29 Postup řešení při užití Laplaceovy transformace

Základní definiční integrál ***přímé Laplaceovy transformace*** je

$$F(p) = \int_0^{\infty} f(t) \cdot e^{-pt} dt$$

Takto definovanou Laplaceovou transformací lze řešit problémy v časové oblasti počínaje časem $t = 0$. Chování systému před tímto časem, tedy jak se systém dostal do výchozího stavu, nelze takto definovanou transformací řešit. Tento stav je popsán počátečními podmínkami řešení.

Aby funkce *f(t)* v integrálu byla integrace schopna, tj. aby existoval obraz, musí být splněny níže uvedené požadavky na funkci f(t):

1. musí být nulová pro záporný čas, tj.

$$f(t) = \{ \begin{matrix} f(t) & pro & t \geq 0, \\ 0 & pro & t < 0, \end{matrix}$$

2. musí být alespoň po částech spojitá,

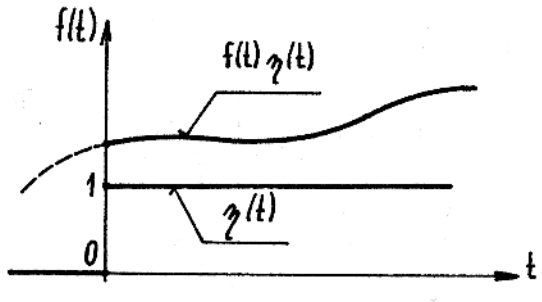
3. musí být funkcí exponencionálního řádu, tj. musí vyhovovat nerovnosti

$$|f(t)| \leq M\, e^{\alpha_0 t}$$

kde $M>0$; $\alpha_0 \in (-\infty, \infty)$, $t \in \langle 0, \infty)$.

První podmínku lze splnit vynásobením dané časové funkce Heavisideovým jednotkovým skokem (viz obr. 30), definovaným vztahem

$$\eta(t) = \{ \begin{matrix} 1 & pro\, t \geq 0 \\ 0 & pro\, t < 0 \end{matrix} .$$



Obr. 30 Použití Heavisideova jednotkového skoku - η(t)

Z obr. 30 vyplývá, že jde prakticky před použitím Laplaceovy transformace o vynásobení původní funkce *f(t)* Heavisideovým skokem η(t)**,** takže dostáváme funkci *f(t) η(t)***,** která vyhovuje první podmínce. Tam, kde nemůže dojit k omylu, zápis *f(t) η(t)* zjednodušujeme tím, že *η(t)* vynecháváme.

Při zápisu označujeme funkce v časové oblasti malými písmeny a říkáme jim ***originály*** *{u(t), y(t)}*, funkce v operátorové oblasti označujeme stejnými velkými písmeny a říkáme jim *obrazy {U(p),Y(p)}*. Vztah mezi originálem a jeho obrazem se nazývá ***korespondence*** a zapisuje se ve tvaru : *f(t) = F(p).*

Zpětná Laplaceova transformace transformuje funkce komplexní proměnné *p* na funkce reálné proměnné *t*.

Nechť *F(p)* je Laplaceova transformace funkce *f(t)* , *t > 0*; potom je zpětná Laplaceova transformace definována integrálem (po uzavřené křivce)

$$f(t) = L^{-1}\{F(p)\} = \frac{1}{2\Pi j}\oint F(p)e^{pt}dp.$$

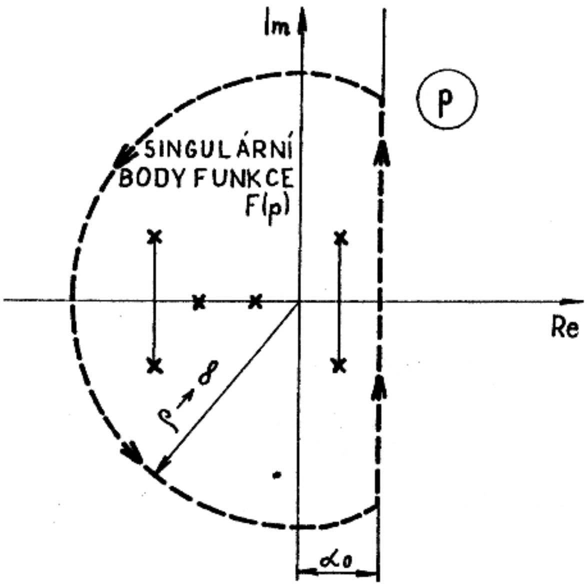
Integrace je provedena. v komplexní rovině a integrační cesta musí být volena tak, aby obepínala všechny póly (singulární body) funkce *F(p).*

Leží-li póly funkce *F(p)* vlevo od přímky $p = \alpha_0$**,** kde α_0 je reálná konstanta větší než nula, tj. v polovině Re $p > \alpha_0$ funkce *F(p)* nemá žádné singulární body (obr. 31), je možné integraci provést podél přímky $p = \alpha_0 + j\omega$. Potom místo integrace v komplexní rovině provedeme prostou integraci podle jedné proměnné ω a místo výše uvedeného vztahu můžeme psát

$$f(t) = L^{-1}\{F(p)\} = \frac{1}{2\Pi j}\int_{\alpha_0 - j\infty}^{\alpha_0 + j\infty} F(p)\cdot e^{pt}dp,$$

kde L^{-1} je operátor zpětné inverzní Laplaceovy transformace,

α_0 – kladná konstanta zvolená tak, aby v polovině Re $p > \alpha_0$ funkce *F(p)* neměla žádné singulární body (obr. 31)



Obr. 31 Singulární body funkce *F(p)*

Základní vlastnosti Laplaceovy transformace

Laplaceova transformace má několik důležitých vlastností, které mohou být použity při výpočtu Laplaceovýeh transformaci funkcí, při řešení lineárních diferenciálních rovnic s konstantnimi koeficienty a při řešení některých úloh řízení dynamických systémů. Je-li F(p) = L{f(t)}, pak platí:

1. Věta o derivování originálu

pro 1. derivaci

$$L\left\{\frac{d\ f(t)}{dt}\right\} = p\,F(p) - f(0),$$

pro n-tou derivaci:

$$L\left\{f^{(n)}(t)\right\} = p^n F(p) - \sum_{i=1}^{n} p^{n-i} \frac{d^{i-1} f(0)}{dt^{i-1}} = p^n F(p) - p^{n-1} f(0) - p^{n-2} f'(0) -$$

$$- p^{n-3} f''(0) - \ldots - f^{(n-1)}(0)$$

2. Věta o integrování originálu

$$L\left\{\int_0^t f(\tau)d\tau\right\} = \frac{1}{p} F(p)$$

3. Věta o počáteční a koncové hodnotě

$$f(0) = \lim_{t \to 0} f(t) = \lim_{p \to \infty} p\,F(p)$$

$$f(\infty) = \lim_{t \to \infty} f(t) = \lim_{p \to 0} p\,F(p)$$

4. Věta o linearitě

$$L\{a_1 f_1(t) + a_2 f_2\} = a_1 F_1(p) + a_2 F_2(p)$$

$$L^{-1}\{b_1 F_1(p) + b_2 F_2(p)\} = b_1 f_1(t) + b_2 f_2(t)$$

5. Věta o posunutí originálu vpravo (zpoždění)

L {f(t-a)} = e^{-ap} *F(p*

kde a ≥ 0

6. Věta a posunutí obrazu, resp. o násobení exponenciální funkcí v časové oblasti

L{e^{-at} *f(t)} = F(p + a).*

L{e^{+at} *f(t)} = F(p - a)*

7. Věta o podobnosti, resp. o změně měřítka

$$L\left\{f\left(\frac{t}{a}\right)\right\} = a\ f(a\,p)$$

$$L^{-1}\left\{F\left(\frac{p}{a}\right)\right\} = a\ f(a\,t)$$

Základní vlastnosti Laplaceovy transformace lze shrnout do následující tabulky:

Tabulka 2 Základní vlastnosti Laplaceovy transformace

Vlastnost	Funkce ve tvaru originálu	Funkce ve tvaru obrazu
Linearita	$f_1(t)+f_2(t)+\ldots+f_n(t)$	$F_1(p)+F_2(p)+\ldots+F_n(p)$
Násobení konstantou	$a.f(t)$	$a.F(p)$
Věta o podobnosti	$f(a.t)$	$\frac{1}{a}\cdot F\left(\frac{p}{a}\right)$
Věta o posunutí vpravo	$f(t-\tau)$	$e^{-p\tau}F(p)$
Derivace	$f^{(n)}(t)$	$p^n.F(p)$ *
Integrace	$\int_0^t f(t)dt$	$\frac{1}{p}\cdot F(p)$ *

Poznámky:

a - konstanta, * - počáteční podmínky musí být nulové

Podmínkou pro použití Laplaceovy transformace je, aby diferenciální rovnice byla lineární, resp. aby platil zákon superpozice (tj. že přírůstky času jsou konstantní).

Abychom nemuseli stále vypočítávat obraz podle definičního integrálu a pak provádět zpětný převod do časové oblasti, jsou zpracovány slovníky LT - stručný slovník LT je uveden v tabulce 3.

Diferenciální rovnici můžeme při nulových počátečních podmínkách použitím věty o obrazu derivace V1 převést na ***přenos soustavy (obrazový přenos)*** - což je obraz diferenciální rovnice při nulových počátečních podmínkách.

$$a_n p^n Y(p) + a_{n-1} p^{n-1} Y(p) + \ldots + a_0 Y(p) = b_m p^m U(p) + \ldots + b_0 U(p)$$

$$Y(p)\left(a_n p^n + \ldots + a_0\right) = U(p)\left(b_m p^m + \ldots + b_0\right)$$

Z čehož obrazový přenos

$$G(p) = \frac{Y(p)}{U(p)} = \frac{b_m p^m + \ldots + b_0}{a_n p^n + \ldots + a_0}$$

Základní rovnice přenosu

$$G(p) = \frac{Y(p)}{U(p)} = \frac{b_0 + b_1 p + \ldots\ldots + b_m p^m}{a_0 + a_1 p + \ldots\ldots + a_n p^n}$$

se často upravuje:

$$G(p)=\frac{\frac{b_0}{a_0}+\frac{b_1}{a_0}p+....+\frac{b_m}{a_0}p^m}{1+\frac{a_1}{a_0}p+....+\frac{a_n}{a_0}p^n}=\frac{\beta_0+\beta_1 p+....+\beta_m p^m}{1+\alpha_1 p+....+\alpha_n p^n}=\frac{\sum_{j=0}^{m}\beta_j p^j}{1+\sum_{i=1}^{n}\alpha_i p^i}$$

V praxi má přenos soustavy často tvar

$$G(p)=\frac{\frac{b_0}{a_0}}{1+\frac{a_1}{a_0}p+....+\frac{a_n}{a_0}p^n}=\frac{K}{1+T_1 p+T_2^2 p^2+....+T_n^n p^n}$$
)

kde K je zesílení soustavy, T_i jsou časové konstanty.

Polynom ve jmenovateli přenosu se nazývá ***charakteristický polynom***. Kořeny tohoto polynomu se nazývají ***póly systému***:

$$a_n\cdot p^n+a_{n-1}\cdot p^{n-1}+...+a_1\cdot p+a_0=a_n(p-p_1)(p-p_2)...(p-p_n)$$

kde *pi (i=1,2.....n)* jsou póly systému.

Kořeny polynomu v čitateli přenosu jsou ***nuly systému***. Čitatel lze psát:

$$b_m\cdot p^m+b_{m-1}\cdot p^{m-1}+...+b_1\cdot p+b_0=b_m(p-n_1)(p-n_2)...(p-n_m)$$

ni (i=1,2...m) jsou nuly systému.

Póly i nuly mohou být buď reálné, komplexně sdružené nebo ryze imaginární. Záporně vzaté převrácené hodnoty reálných pólů a nul jsou časové konstanty.

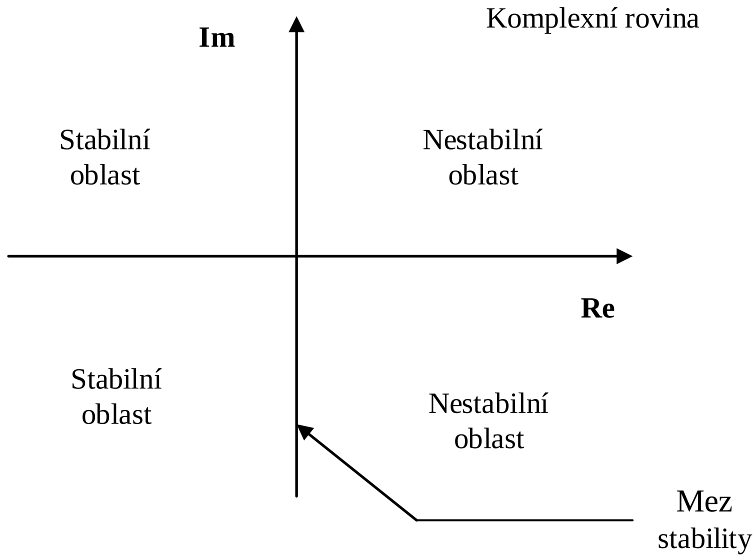
$$T_i=-\frac{1}{p_i}\qquad T_i'=-\frac{1}{n_i}$$

Pak lze přenos psát:

$$G(p)=\frac{b_m}{a_n}\frac{(1+pT_1')(1+pT_2')...+(1+pT_m')}{(1+pT_1)(1+pT_2)...+(1+pT_n)}$$

Na základě znalosti polohy pólů a nul přenosu v komplexní rovině *p*, lze určit vnější chování soustav (obr. 32). Rozložení pólů a nul v komplexní rovině určuje dynamické vlastnosti soustav. U pólů a nul je rozhodující jejich poloha vzhledem k imaginární ose (mez stability). V levé komplexní polorovině jsou *stabilní póly a nuly* (mají zápornou reálnou část), v pravé polorovině jsou *nestabilní póly a nuly* (mají kladnou reálnou část).

Příčinou záporných reálných pólů $(p_i=-\alpha_i,\alpha_i>0)$ je *aperiodický přechodový jev*. Komplexně sdružené póly $(p_i=-\alpha_i\pm j\omega_i,\alpha_i>0)$ způsobují *kmitavý charakter přechodového děje*. Čím jsou stabilní póly dále od imaginární osy, tím je přechodový děj více tlumen.



Obr. 32 Oblast stability spojitých systémů

Tabulka 3 Stručný slovník Laplaceovy transformace

Č.	**Originál x(t)**	**Obraz X(p)**
1	$\delta(t)$	1
2	$\delta(t\text{-}mT)$	e^{-mTp}
3	$\eta(t)$	$\frac{1}{p}$
4	$\eta(t\text{-}mT)$	$\frac{1}{p}e^{-mTp}$
5	t	$\frac{1}{p^2}$
6	$\frac{1}{2}t^2$	$\frac{1}{p^3}$
7	$b^{\mp at}$, $b>0$	$\frac{1}{p \pm a \ln b}$
8	a	$\frac{a}{p}$
9	$e^{\mp at}$	$\frac{1}{p \pm a}$
10	te^{-at}	$\frac{1}{(p+a)^2}$
11	$\frac{1}{2}t^2e^{-at}$	$\frac{1}{(p+a)^3}$
12	$1\text{-}\ e^{-at}$	$\frac{a}{p(p+a)}$

13	$\frac{1}{a}\left[at-(1-e^{-at})\right]$	$\frac{a}{p^2(p+a)}$
14	$(1-at)e^{-at}$	$\frac{p}{(p+a)^2}$
15	$1-(1+at)e^{-at}$	$\frac{a^2}{p(p+a)^2}$
16	$e^{-at}-e^{-bt}$	$\frac{b-a}{(p+a)(p+b)}$
17	$-ae^{-at}+be^{-bt}$	$\frac{(b-a)p}{(p+a)(p+b)}$
18	$1+\frac{be^{-at}+ae^{-bt}}{a-b}$	$\frac{ab}{p(p+a)(p+b)}$
19	$\sin\omega t$	$\frac{\omega}{p^2+\omega^2}$
20	$\cos\omega t$	$\frac{p}{p^2+\omega^2}$
21	$\sinh\omega t$	$\frac{\omega}{p^2-\omega^2}$
22	$\cosh\omega t$	$\frac{p}{p^2-\omega^2}$
23	$e^{-at}\sin\omega t$	$\frac{\omega}{(p+a)^2+\omega^2}$
24	$e^{-at}\cos\omega t$	$\frac{p+a}{(p+a)^2+\omega^2}$

Řešený příklad 6

Vypočítejme Laplaceův obraz Heavisideova jednotkového skoku $\eta(t)$:

$$\eta(t)=\begin{cases}1 & pro\; t\ge 0\\ 0 & pro\; t<0\end{cases}.$$

Řešení:

$$F(p)=L\{1(t)\}=\int_0^\infty f(t)e^{-pt}dt=\int_0^\infty 1e^{-pt}dt=\left[-\frac{1}{p}e^{-pt}\right]_0^\infty=-\frac{1}{p}\left[e^{-\infty}-e^0\right]=\frac{1}{p}.$$

Řešený příklad 7

Dokažme pomoci definičního vztahu Laplaceovy transformace, že platí *L{f´(t)} = p F(p) – f(0)*

Řešení:

$$L\{f'(t)\}=\int_0^\infty f'(t)e^{-pt}dt=\begin{vmatrix} u=e^{-pt}, v'=f'(t) \\ u'=-pe^{-pt}, v=f(t)\end{vmatrix}=\left[e^{-pt}f(t)\right]_0^\infty+\int_0^\infty pe^{-pt}f(t)dt=pF(p)-f(0).$$

Řešený příklad 8

Vypočítejme originál *f(t)* k obrazu *F(p)* s reálnými kořeny polynomu jmenovatele p_1 = -1, p_2 = -2, p_3 = -3:

$$F(p)=\frac{M(p)}{N(p)}=\frac{4p+6}{(p+1)(p+2)(p+3)}.$$

Řešení: Obraz *F(p)* (racionální funkci lomenou) rozložíme v součet parciálních zlomků

$$\frac{4p+6}{(p+1)(p+2)(p+3)}=\frac{A_1}{p+1}+\frac{A_2}{p+2}+\frac{A_3}{p+3}.$$

Konstanty A_1, A_2, A_3 určíme tak, že odstraníme zlomky a srovnáme koeficienty u jednotlivých mocnin komplexní proměnné *p*.

$$A_1=1$$
$$A_2=2$$
$$A_3=-3.$$

Originál získáme zpětnou transformací součtu parciálních zlomků

$$f(t)=L^{-1}\{F(p)\}=L^{-1}\left\{\frac{1}{p+1}+\frac{2}{p+2}-\frac{3}{p+3}\right\}$$

použitím slovníku Laplaceovy transformace v příloze 1:

$f(t) = e^{-t} + 2\,e^{-2t} - 3\,e^{-3t}$

Řešený příklad 9

Najděme originál *f(t)* k danému obrazu *F(p)*, který má v polynomu jmenovatele třínásobný kořen p_1 = -2, jednoduchý kořen p_2 = -3 a nulový kořen p_3 = 0,

$$F(p)=\frac{1}{p(p+2)^3(p+3)}$$

Řešení: Rozklad obrazu F(p) na součet parciálních zlomků použitím Heavisideova rozvoje bude:

$$F(p)=\frac{B_1}{p+2}+\frac{B_2}{(p+2)^2}+\frac{B_3}{(p+2)^3}+\frac{A_1}{p+3}+\frac{A_2}{p}$$

Vypočítáme jednotlivé konstanty odstraněním zlomků a srovnáním koeficientů u jednotlivých mocnin komplexní proměnné *p*:

$$B_3 = -\frac{1}{2}$$

$$B_2 = \frac{1}{4}$$

$$B_1 = -\frac{3}{8}$$

$$A_1 = \frac{1}{3}$$

$$A_2 = \frac{1}{24}$$

Pak

$$F(p) = -\frac{3}{8}\frac{1}{p+2} + \frac{1}{4}\frac{1}{(p+2)^2} - \frac{1}{2}\frac{1}{(p+2)^3} + \frac{1}{3}\frac{1}{p+3} + \frac{1}{24}\frac{1}{p}.$$

Hledaný originál bude

$$f(t) = -\frac{3}{8}e^{-2t} + \frac{1}{4}t\,e^{-2t} - \frac{1}{2.2}t^2 e^{-2t} + \frac{1}{3}e^{-3t} + \frac{1}{24}.$$

Řešený příklad 10

Hledejme řešení diferenciální rovnice
$y''(t) + 5\,y'(t) + 6\,y(t) = 12\,u(t)$

pro kterou platí počáteční podmínky $y(0) = 2$, $y'(0) = 0$ a budící funkce je definována.:

$u(t) = 1$ pro $t \geq 0$

$u(t) = 0$ pro $t < 0$.

Řešení: Použitím věty o derivování originálu a věty o linearitě a skutečnosti, že obraz levé strany diferenciální rovnice se rovná pravé straně této rovnice, dostáváme:

$L\,\{y''(t) + 5\,y'(t) + 6\,y(t)\} = L\{12\,u(t)\}$

$p^2 Y(p) - p\,y(0) - y'(0) + 5\,p\,Y(p) - 5\,y(0) + 6\,Y(p) = 12\,U(p)$

$$Y(p)\left[p^2 + 5p + 6\right] = 2p + 10 + 12\frac{1}{p}$$

$$Y(p) = \frac{2p^2 + 10p + 12}{p(p^2 + 5p + 6)} = \frac{2}{p}$$

Hledané řešení diferenciální rovnice je

$y(t) = L^{-1}\{Y(p)\} = 2$

Řešený příklad 11

Hledejme řešení diferenciální rovnice

$y'(t) + 3\,y(t) = e^{-2t}$

pro počáteční podmínku $y(0) = 0$

Řešení:

$L\{y'(t) + 3y(t)\} = L\{e^{-2t}\}$

$$pY(p) - y(0) + 3Y(p) = \frac{1}{p+2}$$

$$Y(p) = \frac{1}{(p+2)(p+3)}.$$

Použijeme Heavisideova rozvoje

$$Y(p) = \frac{A_1}{p+2} + \frac{A_2}{p+3} = \frac{1}{p+2} - \frac{1}{p+3},$$

kde

$$A_1 = 1$$
$$A_2 = -1.$$

Hledané řešení je

$y(t) = L^{-1}\{Y(p)\} = e^{-2t} - e^{-3t}.$

Rozdělení základních lineárních dynamických soustav

Lineární dynamické soustavy lze třídit podle různých hledisek. Velmi důležitým hlediskem je vlastnost ustáleného stavu, a to, zda existuje (a je nenulový, příp. nulový) nebo neexistuje Vlastnost ustáleného stavu se velmi dobře posuzuje podle průběhu přechodové charakteristiky $h(t)$ pro $t \to \infty$. Podle tohoto hlediska dělíme soustavy na:

- ***proporcionální (dřívější termín statické)***
- ***integrační (dřívější termín astatické)***
- ***derivační***

Proporcionální soustavy

Řád diferenciální rovnice popisující soustavu vyjadřuje řád soustavy. Proporcionální soustavy se při vychýlení z ustáleného stavu samy ustálí na nové hodnotě ustáleného stavu, přičemž tento stav je nenulový. Proporcionální soustavy se vyznačují tím, že ve jmenovateli ani v čitateli přenosu $G(p)$ nelze vytknout komplexní proměnnou p. V diferenciální rovnici popisující proporcionální soustavu je koeficient a_0 nenulový. Přenosy základních proporcionálních členů mají tvar:

1. Soustava nultého řádu (proporcionální člen bez setrvačnosti, ideální proporcionální člen)

$$a_0\, y(t) = b_0\, u(t)$$

$$G(p) = \frac{Y(p)}{U(p)} = \frac{b_0}{a_0} = k$$

kde k je zesílení soustavy (koeficient přenosu). Jeho fyzikální rozměr je dán poměrem fyzikálních rozměrů výstupní $y(t)$ a vstupní $u(t)$ veličiny. Pro $|k| > 1$ jde o zesílení a pro $|k| < 1$ o tlumení.

Příklady: zesilovače (proporcionální regulátor), převodovky, potrubí s kapalinami.

Obr. 29 Přechodová charakteristika soustavy 0. řádu

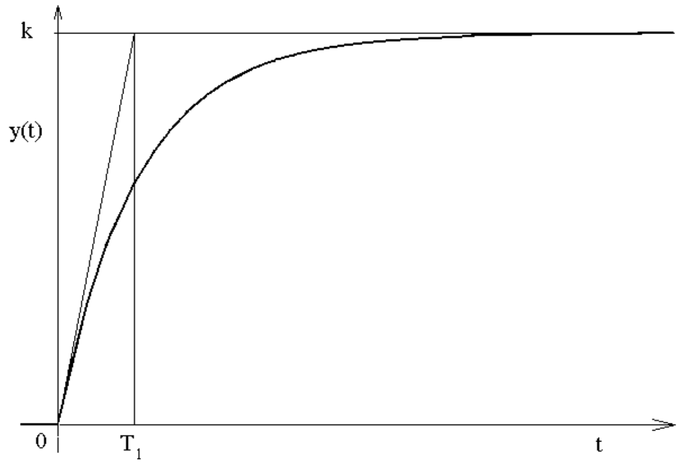
2. Soustava prvního řádu (proporcionální člen se setrvačností 1. řádu)

$$a_1 y'(t) + a_0 y(t) = b_0\, u(t)$$

$$G(p) = \frac{Y(p)}{U(p)} = \frac{b_0}{a_1\, p + a_0} = \frac{k}{T\, p + 1}$$

kde je $k = \frac{b_0}{a_0}$ je zesílení soustavy (koeficient přenosu), $T = \frac{a_1}{a_0}$ je časová konstanta (s fyzikálním rozměrem čas).

Příklady: tlakové nádoby, průtok plynu, elektrické obvody s odpory a kapacitami, povrchová teplota materiálu, nádrže při regulaci hladiny, regulace otáček motorů.



Obr. 30 Přechodová charakteristika soustavy 1. řádu

3. Soustava druhého řádu (proporcionální člen se setrvačností 2. řádu)

$$a_2 y''(t) + a_1 y'(t) + a_0 y(t) = b_0 u(t)$$

$$G(p) = \frac{Y(p)}{U(p)} = \frac{b_0}{a_2 p^2 + a_1 p + a_0} = \frac{k}{T_2 p^2 + T_1 p + 1}$$

kde k je zesílení soustavy (koeficient přenosu), T_1, T_2 jsou časové konstanty.

O průběhu přechodové charakteristiky rozhoduje rozložení pólů a nul v komplexní rovině. Póly systému získáme řešením charakteristické rovnice (kvadratického trojčlenu ve jmenovateli přenosu):

$$p^2 + \frac{T_1}{T_2} p + \frac{1}{T_2} = 0$$

$$p_{1,2} = -\frac{T_1}{2T_2} \pm \sqrt{\left(\frac{T_1}{2T_2}\right)^2 - \frac{1}{T_2}}$$

Průběh přechodové charakteristiky závisí na hodnotě diskriminantu D rovnice (89):

1) $D < 0$ (kořeny komplexně sdružené) - přechodová charakteristika je kmitavá (periodická),
2) $D = 0$ (dvojnásobný kořen) – přechodová charakteristika je na mezi aperiodicity,
3) $D > 0$ (dva reálné záporné kořeny) – přechodová charakteristika je nekmitavá (aperiodická).

Jsou-li kořeny charakteristické rovnice komplexně sdružené (kmitavá soustava), pak je možno přenos vyjádřit ve tvaru:

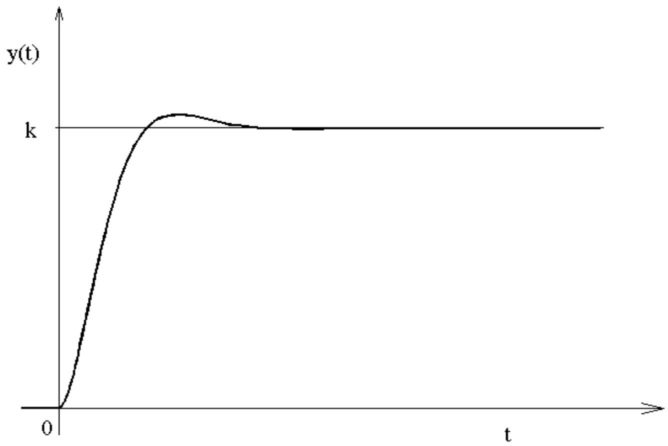
$$G(p) = \frac{k}{T_0^2 p^2 + 2\xi T_0 p + 1}.$$

kde T_0 je perioda kmitání a ξ je koeficient tlumení.

Příklady:

Apriodický průběh mají: soustavy tepelné, výměníky tepla, pece, nádrže, tlakové nádoby, regulace otáček a všechny soustavy složené ze dvou členů prvního řádu.

Kmitavý průběh mají: soustavy obsahující setrvačné hmoty (hmotnost na pružině), elektrické obvody současně obsahující odpory, indukčnosti a kapacity (oscilační obvody).



Obr. 31 Přechodová charakteristika kmitavé soustavy 2. řádu

4. Soustava s dopravním zpožděním (ideálně zpožďující člen)

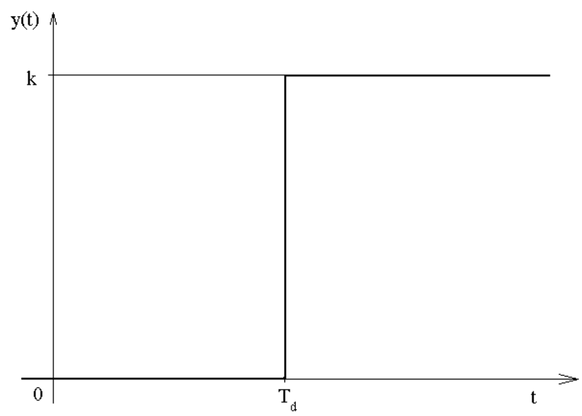
$$a_0\, y(t) = b_0\, u(t - T_d)$$

$$G(p) = \frac{b_0}{a_0} e^{-pT_d} = k\, e^{-pT_d}$$

kde T_d je dopravní zpoždění (s fyzikálním rozměrem čas).

Člen dopravního zpoždění se v časové oblasti vyznačuje tím, že časovou odezvu nezmění, ale pouze zpozdí o dopravní zpoždění T_d.

Příklady: dopravníky, řízení kontinuálních válcovacích stolic, vrstvení tekutými materiály (polévání filmové podložky emulsí).



Obr. 32 Přechodová charakteristika soustavy s dopravním zpožděním

Integrační (astatické) soustavy

U integračních soustav klidový ustálený stav *h(∞)* na přechodové charakteristice *h(t)* neexistuje, a proto u nich neexistuje ani statická charakteristika. Integrační soustavy se po vychýlení z ustáleného stavu bez působení regulátoru již neustálí v nové hodnotě ustáleného stavu. Prakticky to znamená, že pro každou nenulovou ustálenou hodnotu vstupní veličiny výstupní veličina roste (nebo klesá) až na hodnotu danou fyzikálním omezením. Integrační soustavy se vyznačují tím, že ve jmenovatelích jejich přenosů *G(p)* lze vytknout komplexní proměnnou *p*. V diferenciální rovnici popisující integrační soustavu je koeficient a_0 = 0. Přenosy základních integračních členů mají tvar:

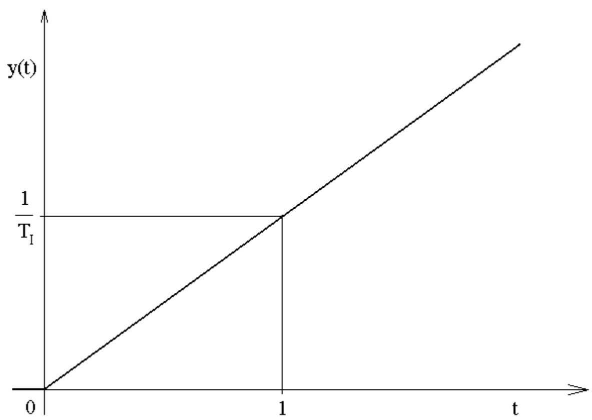
1. Soustava prvního řádu (integrační člen bez setrvačnosti, ideální integrační člen)

$$a_1 y'(t) = b_0 u(t)$$

$$G(p) = \frac{Y(p)}{U(p)} = \frac{b_0}{a_1 p} = \frac{k_1}{p} = \frac{1}{T_I\, p} \qquad (94)$$

kde $k_1 = \frac{b_0}{a_1}$ je koeficient přenosu s fyzikálním rozměrem daným poměrem fyzikálních rozměrů výstupní *y(t)* a vstupní *u(t)* veličiny dělený časem, T_I je integrační časová konstanta ($T_I = \frac{1}{k_1}$).

Příklady: integrační regulátor, řízení vozidel, plnění velkých zásobníků plynem, zásobníky sypkých hmot, nádrže s nuceným přítokem a odtokem, kondenzátory.



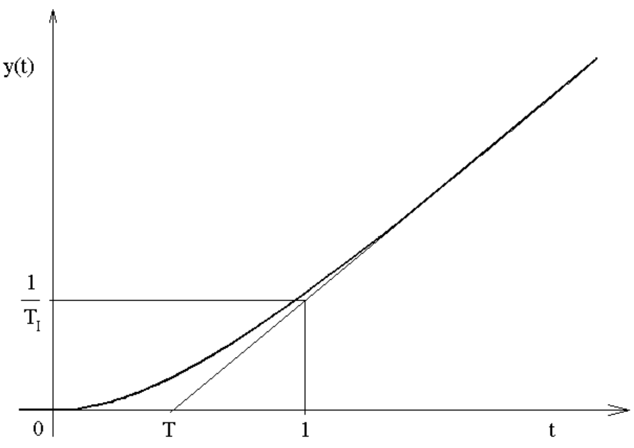
Obr. 33 Přechodová charakteristika integrační soustavy bez setrvačnosti

2. Soustava druhého řádu (integrační člen se setrvačností 1. řádu, reálný integrační člen)

$$a_2 y''(t) + a_1 y'(t) = b_0 u(t)$$

$$G(p) = \frac{b_0}{p(a_2 p + a_1)} = \frac{k_1}{p(T_1 p + 1)} = \frac{1}{T_I\, p(T_1 p + 1)}$$

kde $k_1 = \frac{b_0}{a_1}$ a $T_1 = \frac{a_2}{a_1}$.



Obr. 34 Přechodová charakteristika integrační soustavy se setrvačností 1. řádu

V literatuře se užívá termín ***řád astatismu***. Udává počet integračních členů. Např. integrační soustava $q+n$-tého řádu se setrvačností n-tého řádu s řádem astatismu q má přenos:

$$G(p) = \frac{b_0}{p^q\left(a_n p^n + a_{n-1} p^{n-1} \ldots + a_1\right)}$$

Derivační soustavy

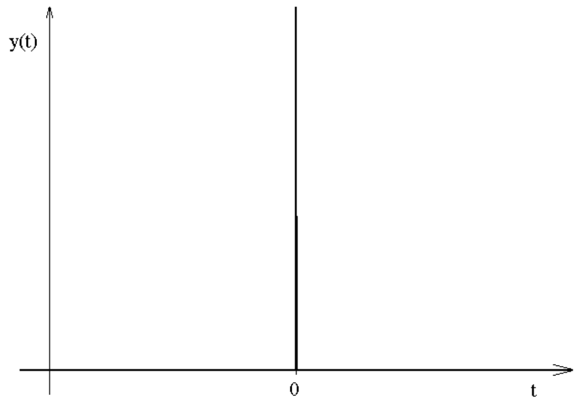
U derivačních soustav sice ustálený stav $h(\infty)$ na přechodové charakteristice $h(t)$ existuje, ale je nulový, tzn. statická charakteristika je triviální $y(\infty)=0$. Přenosy derivačních soustav se vyznačují tím, že v jejich čitatelích lze vytknout komplexní proměnnou p. Přenosy základních derivačních členů mají tvar:

1. Derivační soustava bez setrvačnosti (ideální derivační člen)

$$a_0\, y(t) = b_1\, u'(t)$$

$$G(p) = \frac{Y(p)}{U(p)} = \frac{b_1}{a_0} \cdot p = T_D \cdot p$$

kde $T_D = \frac{b_1}{a_0}$ je je koeficient přenosu (derivační časová konstanta) s fyzikálním rozměrem daným poměrem fyzikálních rozměrů výstupní $y(t)$ a vstupní $u(t)$ veličiny násobený časem. Přechodovou charakteristikou je Diracův impuls. Ideální derivační člen je fyzikálně nerealizovatelný (m > n).



Obr. 35 Přechodová charakteristika ideální derivační soustavy

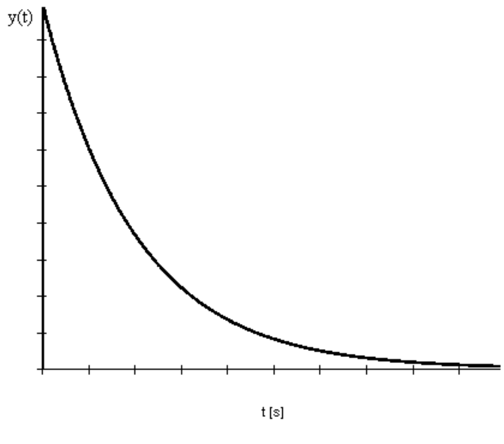
2. Derivační soustava se setrvačností 1. řádu (reálný derivační člen)

$$a_1 y'(t) + a_0 y(t) = b_1 u'(t)$$

$$G(p) = \frac{Y(p)}{U(p)} = \frac{b_1 p}{a_1 p + a_0} = \frac{b_1 p}{T p + 1} = \frac{T_D p}{T p + 1}$$

kde $T_D = \frac{b_1}{a_0}$ je koeficient přenosu a $T = \frac{a_1}{a_0}$ je časová konstanta.

Příklad: derivační regulátor, elektrické obvody s odpory a kapacitami nebo s odpory a indukčnostmi.



Obr. 36 Přechodová charakteristika reálné derivační soustavy

Σ Shrnutí pojmů

Matematický popis dynamických systémů, popis lineární diferenciální rovnicí, základní vlastnosti Laplaceovy transformace, popis přenosem v Laplaceově transformaci, rozdělení základních lineárních dynamických soustav.

Otázky

1. Uveďte příklady matematického popisu spojitých lineárních soustav.
2. Jakým způsobem popisujeme statické s dynamické vlastnosti dynamických systémů?
3. Popište základní vlastnosti Laplaceovy transformace.
4. Definujte přenos v Laplaceově transformaci.
5. Proveďte klasifikaci základních lineárních dynamických soustav

7. Logické systémy

Čas ke studiu: 3 hodiny

Cíl Po prostudování tohoto odstavce budete umět

- definovat pojem logická proměnná a logická funkce
- popsat základní logické funkce
- definovat a vysvětlit základní zákony Booleovy algebry
- popsat způsoby realizace a minimalizace logických funkcí

Výklad

Logická proměnná, logická funkce

Při vyhodnocování stavů technologického procesu mnohdy dostačuje zjistit, zda nějaká činnost nastala nebo nenastala, např. motor se točí - netočí, bylo dosaženo určité teploty, ventil je otevřen - zavřen. Hodnoty mezi těmito dvěma stavy nás nezajímají. Informace o této skutečnosti nabývá pouze dvou hodnot. Výhodou je, že zpracování takové informace je možno provést jednoduššími a spolehlivějšími prostředky než při zpracování spojitých signálů.

Vstupní členy převádějí (zpravidla spojité) vstupní veličiny na nespojitý výstupní signál, který nabývá pouze dvou hodnot. Jsou to např. kontaktní nebo bezdotykový snímač polohy, kontaktní manometr, kontaktní teploměr, různá tlačítka, spínače apod.

Výstupní členy zpracovávající takovou dvouhodnotovou informaci a působí jako akční členy v navazujících obvodech. Jsou to např. relé, stykače, elektromagnetické spojky, elektromagnety apod.

Filozofická disciplína *logika* - nauka o výrocích a vazbách mezi nimi - přiřazuje takovým informacím označovaným jako ***logické proměnné*** čísla 0 a 1 nebo též 0 a I. Říkáme, že logická proměnná má hodnotu logické nuly nebo logické jedničky. Význam logických hodnot je

0 - výrok neplatí, činnost, signál neexistuje, obvod nevede....

1 - výrok platí, činnost, signál existuje, obvod vede....

Logická proměnná vyjadřuje pouze dva stavy. Je-li logických proměnných n, pak lze jimi vyjádřit 2^n různých stavů.

Jestliže jednotlivým logickým proměnným přisoudíme jednotlivé řády binárního čísla, pak dekadický ekvivalent tohoto binárního čísla označujeme jako stavový index s.

Vztah mezi logickými proměnnými je určen tzv. logickou funkcí.

Logická funkce je předpis, který přiřazuje kombinacím hodnot jedné nebo více vstupních logických proměnných hodnotu výstupní proměnné.

Jedním ze způsobů vyjádření logické funkce je tzv. ***pravdivostní tabulka***. V její levé části jsou uvedeny všechny možné kombinace hodnot vstupních proměnných, v pravé části je těmto kombinacím přiřazena výstupní hodnota logické funkce.

Základní logické funkce

Pro označení vstupních proměnných obvykle užíváme malá písmena ze začátku abecedy (budeme užívat písmena *a*, *b*), pro výstupní proměnné malá písmena z konce abecedy (užijeme písmeno *y*).

Funkce jedné proměnné, negace

Nejsnáze lze demonstrovat logické funkce na případu funkcí jedné vstupní proměnné *a*. Pravdivostní tabulka této funkce bude mít na levé straně pouze jeden sloupec. Hodnotám této jediné nezávislé proměnné lze přiřadit výstupní hodnoty čtyřmi způsoby, tedy existují čtyři logické funkce jedné proměnné y_1 až y_4, jejichž pravdivostní tabulky shrneme do společné tabulky s jediným vyjádřením hodnot vstupní proměnné *a*.

a	y_1	y_2	y_3	y_4
0	0	0	1	1
1	0	1	0	1

Z těchto čtyř funkcí je nejdůležitější funkce y_3, která přiřazuje výstupu opačnou hodnotu, než má vstup. Tuto funkci nazýváme *negace***.**

Slovní označení: ***negace, inverse, "non"***

$y = \bar{a}$ tedy $\bar{1} = 0$, $\bar{0} = 1$

Pravdivostní tabulka:

a	*y*
0	1
1	0

Funkce dvou proměnných

Počtu *n* vstupních proměnných lze obecně přiřadit 2^{2n} logických funkcí. Dvě vstupní proměnné dávají čtyři kombinace vstupních hodnot, kterým lze přiřadit 16 různých logických funkcí. Nejdůležitější z nich jsou logický součin a součet.

Slovní označení: ***logický součin, "i", "AND", konjunkce, průnik***

$y = a \cdot b$ tedy 1 . 1 = 1, 1 . 0 = 0, 0 . 0 = 0

Pravdivostní tabulka:

a	*b*	*y*
0	0	0
0	1	0
1	0	0
1	1	1

Slovní označení: ***logický součet, "nebo", "OR", disjunkce, sjednocení***

$y = a + b$ tedy 1 + 1 = 1, 1 + 0 = 1, 0 + 0 = 0

Pravdivostní tabulka:

a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

Funkce logického součtu i logického součinu lze samozřejmě definovat pro libovolný počet vstupních proměnných.

Složené logické funkce

Mezi šestnáct možných logických funkcí dvou proměnných patří i následující dvě funkce, jejichž hlavní význam je ve skutečnosti, že libovolnou logickou funkci lze realizovat výhradním užitím členů realizujících jednu z nich. Můžeme je ale definovat rovněž jako funkce složené z výše uvedených tří funkcí.

Negovaný logický součin- funkce Shefferova

$$y = \overline{a \cdot b}$$

Pravdivostní tabulka:

a	b	$a.b$	$y = \overline{a \cdot b}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Negovaný logický součet - funkce Pierceova

$$y = \overline{a + b}$$

Pravdivostní tabulka:

a	b	$a+b$	$y = \overline{a + b}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

Operace s logickými proměnnými, funkcemi

Pro operace s logickými funkcemi lze definovat tzv. *logickou algebru*, tedy soubor axiomů, který musí být:

- *konzistentní* (bezesporný), tj. že při správném postupu nelze odvodit odporující si výroky,
- *úplný*, tj. že přidáním jakéhokoli dalšího pravidla, výroku, axiomu by se porušila konzistentnost této algebry.

Nejznámější a v technické praxi nejužívanější logickou algebrou je tzv. Booleova algebra nazvaná po významném irském matematikovi a logikovi Georgu Booleovi (1815 - 1864), která se opírá o tři základní operace:

- *negaci,*
- *logický součin (konjunkci*),
- *logický součet (disjunkci).*

Základní axiomy Booleovy algebry

Jedem z možných systémů definujících Booleovu algebru zavedl v roce 1904 Američan E. V. Huntington.

Booleovou algebrou pak nazýváme každou množinu B obsahující dva různé prvky 0 a 1 a dále prvky x, y, z, ...a v níž jsou definovány operace součtu ($x+y$), součinu (xy) a negace ($\overline{x}$) tak, že platí následující soubor axiomů:

1. <u>Vnitřní zákony kompozice:</u>

Jestliže x a y jsou prvky množiny B, pak také $x+y$ a xy jsou prvky této množiny.

2. <u>Zákony absorpce:</u>

V množině B je význačný prvek 0 takový, že $x + 0 = x$ je splněno pro libovolný prvek x množiny B. Obdobně existuje význačný prvek 1, u něhož platí $x.1 = x$.

3. <u>Komutativní zákony:</u>

Pro prvky x a y množiny B je vždy splněno $x + y = y + x$ a současně $xy = yx$.

4. <u>Distributivní zákony:</u>

Pro prvky x, y a z z množiny B je vždy splněno $x + (yz) = (x + y)(x + z)$ a současně $x(y + z) = xy + xz$.

5. <u>Zákony vyloučení třetího:</u>

Ke každému prvku x z množiny B se vyskytuje prvek $\overline{x}$, pro který platí $x\overline{x} = 0$ a současně $x + \overline{x} = 1$.

6. <u>Jsou alespoň dva takové prvky x a y z množiny B, že platí x ≠ y.</u>

Z těchto axiomů lze odvodit veškerá pravidla pro operace s logickými funkcemi a proměnnými prováděnými užitím Booleovy algebry.

Logický výraz, tzv. Booleův se tedy skládá ze symbolů 0 a 1, z písmenného označení logických proměnných x, y, z, ...a ze symbolů operací Booleovy algebry, tedy součinu, součtu a negace.

Pro operace platí následující pravidla:

- operace se provádí v pořadí nejprve operace negace, další operace logického součinu a nakonec operace logického součtu,
- jestliže jsou ve výrazu užity závorky, provádí se nejprve operace uzavřené v nejvnitřnějších závorkách,
- při negaci složitějšího výrazu znak negace nad tímto výrazem nahrazuje uzavření tohoto výrazu do závorek, což je nutno respektovat při aplikaci předchozího pravidla.

Souhrn pravidel Booleovy algebry

Pro operace v takto definované Booleově algebře lze odvodit následující základní pravidla zahrnující rovněž její základní axiomy:

1. Zákon agresivnosti a neutrálnosti prvků 0 a 1

$x + 1 = 1 \qquad x + 0 = x \qquad x.0 = 0 \qquad x.1 = x$

2. Komutativní zákon

$x + y = y + x \qquad xy = yx$

3. Asociativní zákon

$x + (y + z) = (x + y) + z \qquad x(yz) = (xy)z$

4. Distributivní zákon

$x + (yz) = (x + y)(x + z) \qquad x(y + z) = xy + xz$

5. Zákony absorpce

$x + x = x \qquad xx = x$

$x + xy = x \qquad x(x + y) = x$

6. Zákony absorpce negace

$x + \overline{x}y = x + y \qquad x\left(\overline{x} + y\right) = xy$

$\overline{x} + xy = \overline{x} + y \qquad \overline{x}\left(x + y\right) = \overline{x}y$

7. Zákon dvojité negace

$\overline{\overline{x}} = x$

8. Zákon vyloučení třetího

$x + \overline{x} = 1 \qquad x\overline{x} = 0$

9. De Morganovo pravidlo (pravidla o vytvoření negace)

$\overline{x + y + z} = \overline{x} \cdot \overline{y} \cdot \overline{z} \qquad \overline{x \cdot y \cdot z} = \overline{x} + \overline{y} + \overline{z}$

Negace součtu proměnných je rovna součinu negovaných proměnných. Negace součinu proměnných je rovna součtu negovaných proměnných. De Morganovo pravidlo platí pro libovolný počet logických proměnných.

Realizace logických funkcí

Logických funkcí používáme k řízení technologických procesů. Jedná se jak o systémy zcela jednoduché, např. dvoutlačítkové ovládání motoru, až po vrchol současného logického řízení - číslicový počítač - který veškeré funkce včetně matematických operací s čísly - provádí užitím základních logických funkcí.

Logické funkce jsou realizovány ***logickými obvody***. Ty mohou mít různou technickou podstatu, lze je realizovat mechanicky (např. dveřní zámek tvoří funkci logického součinu), pneumaticky, elektricky. V současné době převládá realizace elektronickými polovodičovými obvody, tzv. integrovanými obvody, které mají na jedné křemíkové destičce - substrátu - integrovány elektronické obvody realizující tyto logické funkce. Nejsložitějším z těchto obvodů je mikroprocesor - srdce všech moderních počítačů. Nejrozsáhlejší jsou polovodičové paměti.

Logické obvody (funkce) dělíme na:

1) **kombinační** - hodnota výstupních veličin závisí jen na kombinaci vstupních veličin.

2) **sekvenční -** hodnota výstupních veličin závisí jednak na kombinaci vstupních veličin a dále na předchozím stavu (např. logické automaty pro řízení výrobních linek, automatické pračky apod.). Tyto obvody musí vždy obsahovat vnitřní proměnné (paměti). Pro realizaci sekvenčních obvodů se užívá stejných prvků jako pro obvody kombinační. Informace o předchozím stavu systému se získávají zavedením výstupních veličin na vstupy zpracovávajících členů současně se vstupními veličinami. Sekvenční logické obvody mohou být:

    - *synchronní* - všechny změny v logickém obvodu probíhají současně. Změny jsou řízeny synchronizačními impulsy.
    - *asynchronní* - stav obvodu se mění ihned po změně vstupu, práce obvodu není synchronizována.

Logické funkce lze realizovat technickými prostředky, které jsou založeny na následujících technických prvcích:

- **mechanická relé**
- **integrované obvody**
- **programovatelné logické automaty (PLC)**

Každému z těchto prvků odpovídá specifický způsob znázorňování schéma zapojení těchto prvků. Reléové obvody se znázorňují v liniových kontaktních schématech. Elektronické logické prvky využívají blokových schémat a funkce programový logických automatů je popsána některou formou dokumentace software. Při realizaci logických funkcí vycházíme zpravidla z minimalizovaného tvaru logické funkce.

Realizace logických obvodů užitím relé

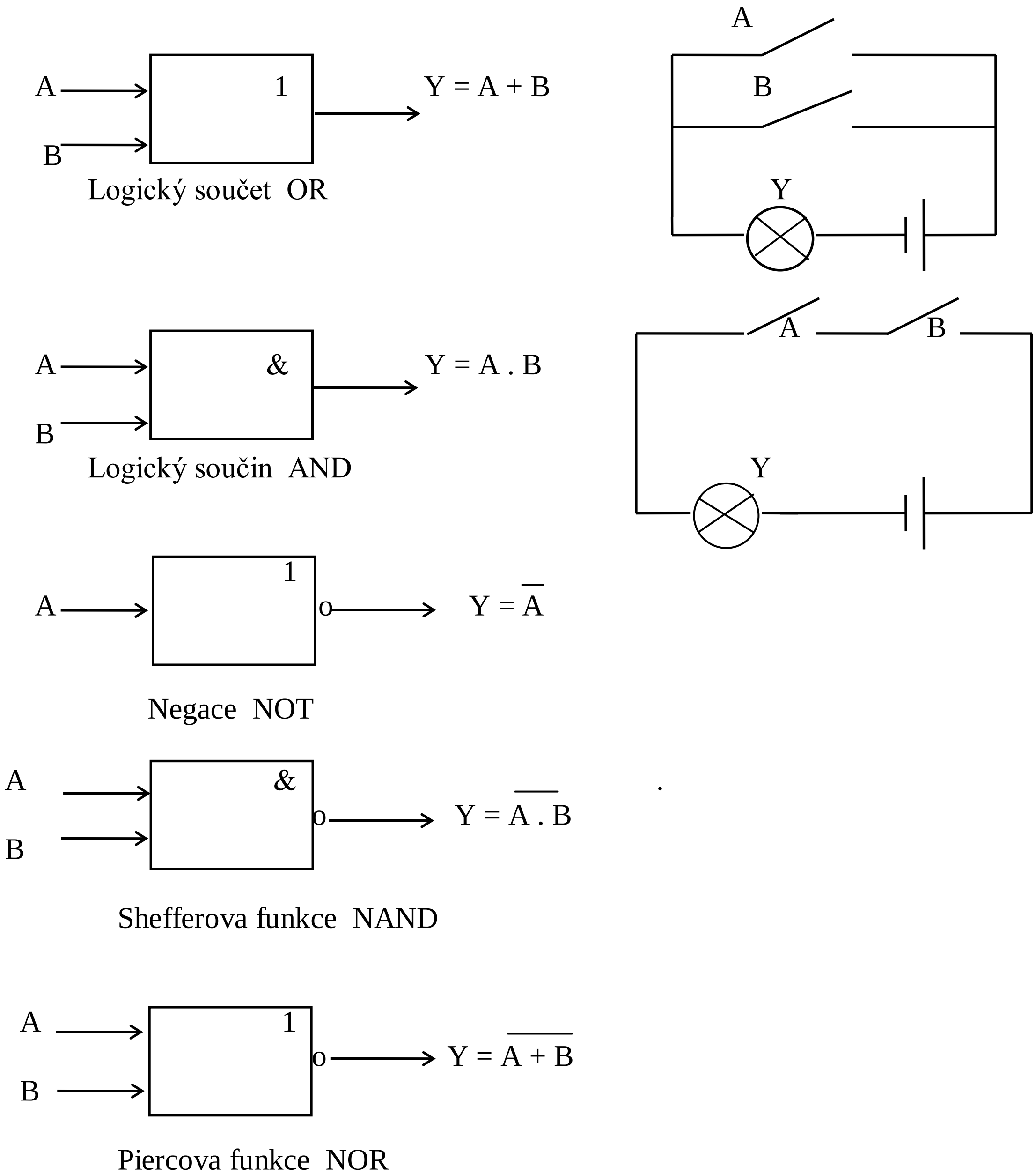
Relé je přístroj obsahující elektromagnet, který ovládá spínání kontaktů. Kontakty jsou dvojího druhu, tzv. pracovní (spínací), které jsou sepnuty tehdy, je-li cívka relé pod proudem, a dále klidové (rozpínací), které jsou sepnuty v bezproudém stavu cívky a po připojení proudu se rozepnou.

Klidové a pracovní kontakty mají i různé ovládací prvky, jako jsou tlačítka, koncové spínače apod.

Realizace užitím logických členů, tzv. hradel

V současné době jsou k disposici elektronické prvky, tzv. logické integrované obvody. Na křemíkové destičce, tzv. čipu, jsou vytvořeny polovodičové obvody realizující různé logické funkce, tzv. hradla. Propojením vývodů těchto základních obvodů lze vytvořit složitější funkce. Jestliže chceme použít integrované obvody realizující negované logické součiny, upravíme logický výraz do tvaru vyjadřující funkci pomocí negovaných logických součinů. Výsledný tvar je pak návodem pro realizaci obvodu.

Schematické znázornění základních logických funkcí pomocí hradel a relé je uvedeno na obr. 37:



Obr. 37 Realizace základních logických funkcí pomocí hradel a relé.

Realizace logických funkcí programovatelnými logickými automaty

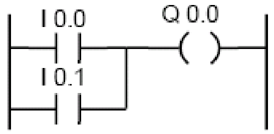
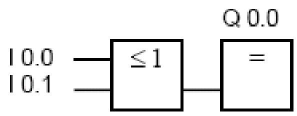
Programovatelné logické automaty (PLC) představují speciální počítač. Jejich funkce se řídí programem, což je posloupnost instrukcí provádějící aritmeticko-logické operace a přesuny mezi svou operační pamětí, aritmeticko- logickou jednotkou (ALU) a vnějším prostředím.

V praxi se používají tři způsoby programování logických automatů pro kombinační logické úlohy. Jsou to jazyky:

- liniových, resp. kontaktních schémat
- blokových schémat logické funkce
- symbolických instrukcí

Prví dva způsoby využívají grafiku, zatímco třetí je založen textově. V tabulce 1.6 je ukázka uvedených způsobů programování logických automatů na příkladu logického součtu dvou logických proměnných. Jména vstupních logických proměnných jsou ve skutečnosti označení symbolických adres I 0.0 a I 0.1, důležitých pro obsazení paměti počítače. Výsledek je uložen na adrese Q 0.0. Všechny tři proměnné jsou typu BOOL (Boolean). U blokového schématu je použit blok výsledku s adresou nad jeho značkou. Jazyk symbolických instrukcí používá posloupnost příkazů v řádcích, ve kterých na prvním místě je typ operace a na druhém místě operand, což je u příkazu použitých v příkladu adresa, ale neplatí to obecně. V příkladu je typ operace označen symbolem O (OR), tj. logický součet. Pro logický součin se používá zkratka A (AND). Symbol = označuje uložení výsledků na adresu, která je operandem tohoto příkazu.

Tabulka 3 Způsoby programování logických automatů

<table>
<tr><th>Jazyk kontaktních schémat</th><th>Jazyk blokových schémat</th><th>Jazyk symbolických instrukcí</th></tr>
<tr><td></td><td></td><td>O I 0.0
O I 0.1
= Q 0.0</td></tr>
</table>

Vyjádření logických funkcí

Základní grafické vyjádření logických funkcí představují ***pravdivostní tabulky***, ***logické mapy*** a ***logické výrazy***. Tyto formy zápisu se užívají pro úvodní operace zápisu a zpracování logických funkcí, logických výrazů tak, abychom získali konečnou formu výrazu vhodnou pro jeho realizaci v logickém řízení. Tato forma je v dalším zpravidla minimalizována, tj. hledáme takový tvar logického výrazu, aby bylo pro jeho realizaci možno použít minimálního počtu prvků.

Pravdivostní tabulka

Definujme např. logickou funkci tří proměnných $y = f$ (a, b, c). Nechť je tato funkce popsána pravdivostní tabulkou. V prvých třech sloupcích pravdivostní tabulky jsou zapsány všechny kombinace hodnot vstupních proměnných, ve čtvrtém pak hodnota funkce odpovídající příslušné kombinaci vstupních proměnných. Někdy pro orientaci uvádíme v pravdivostní tabulce i stavový index *s*.

Pravdivostní tabulka zadané funkce

s	a	b	c	y
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	1
6	1	1	0	0
7	1	1	1	0

Touto pravdivostní tabulkou je příslušná logická funkce plně zadána.

Mapy

Logickou funkci lze pro přehlednost zadávat mapou. Mapa je grafickým znázorněním Booleovy funkce, která vyjadřuje vztah závislé a nezávislé proměnné. Pro *n* vstupních proměnných je počet kombinací jejich hodnot 2^{n}. Mapa je tvořena obdélníkem nebo čtvercem rozdělným na políčka, přičemž každému políčku odpovídá jedna kombinace hodnot vstupních proměnných. Do políčka pak zapisujeme hodnotu logické funkce odpovídající této kombinaci vstupních veličin.

0 **1**	1 **0**	2 **0**	3 **0**
4 **1**	5 **1**	6 **1**	7 **1**
8 **1**	9 **1**	10 **0**	11 **1**
12 **1**	13 **1**	14 **0**	15 **1**

Velmi často se do mapy zapisují pouze hodnoty funkce, ve kterých nabývá funkce hodnoty 1.

Logické výrazy

Užívají se dva základními tvary zápisu logické funkce logickým výrazem:

- *úplná disjunktivní normální forma (ÚDNF),* tedy součet součinů základních proměnných nebo jejich negací,
- *úplná konjunktivní normální forma (ÚKNF),* tedy součin součtů základních proměnných nebo jejich negací.

Při přepisu funkce zadané pravdivostní tabulkou do tvaru logického výrazu lze postupovat víceméně mechanicky.

Přepis do tvaru součtu součinů (ÚDNF) provedeme tak, že vyhledáváme ty kombinace vstupních proměnných, pro které má výstupní proměnná hodnotu 1. Pro každou takto nalezenou kombinaci napíšeme takový součin vstupních proměnných, resp. jejich negací, aby tento součin měl právě hodnotu 1. Znamená to, že v případě, že vstupní proměnná má v daném řádku hodnotu 1, zapíšeme tuto proměnnou přímo, pokud má vstupní proměnná¨hodnotu 0, zapíšeme do výrazu negaci této vstupní proměnné. Součet takto vytvořených součinů je logickým výrazem dané logické funkce.

Přepíšeme logickou funkci zadanou tabulkou do logického výrazu ve tvaru ÚDNF. V tabulce postupně shora vyhledáváme ty řádky, v nichž je hodnota y = 1

člen odpovídající stavovému indexu

s = 0 4 5

$$y = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + a\bar{b}c \qquad (1)$$

Při přepisu do tvaru součinu součtů (ÚKNF) naopak vyhledáváme ty řádky, kde je hodnota funkce y = 0 a do jednotlivých součtů zapisujeme podmínky odpovídající nulové hodnotě tohoto součtu. Tedy naopak proti minulému postupu musíme zapsat přímou proměnnou v případě, že tato proměnná má v daném řádku hodnotu 0 a negaci této proměnné, jestliže má tato proměnná hodnotu 1. Součinem těchto podmínek dostaneme výslednou výraz určující podmínky nulové hodnoty této funkce. Pro funkci zadanou tabulkou:

člen odpovídající stavovému indexu

s = 1 2 3 6 7

$$y = (\bar{a} + b + c)(a + \bar{b} + c)(\bar{a} + \bar{b} + c)(a + \bar{b} + \bar{c})(\bar{a} + \bar{b} + \bar{c}) \qquad (2)$$

Minimalizace logických výrazů

Zadání logické funkce některým z uvedených způsobů není pro konečnou realizaci vhodné. Proto musíme získaný logický výraz zjednodušit, tzv. minimalizovat, a případně upravit do takového tvaru, aby byl realizovatelný zvolenými prvky.

Zápis logické funkce ve tvaru pravdivostní tabulky je poměrně přehledný, ale není vhodným výchozím tvarem pro minimalizaci výsledného logického výrazu.

Minimalizace podle pravidel Booleovy algebry

Logický výraz (1) můžeme upravovat podle pravidel Booleovy algebry. Zde záleží na zkušenostech, zda dosáhneme minimálního vyjádření zadané logické funkce. Nicméně jsou tyto postupy vhodné pro vytvoření takového výsledného výrazu, který umožňuje realizovat logickou funkci vybranými technickými prostředky.

Upravme výraz (1), přitom se budeme odvolávat na pravidla Booleovy algebry:

$$y = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + a\bar{b}c = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c + \bar{a}\bar{b}c + a\bar{b}c = \bar{a}\bar{b}(\bar{c} + c) + (\bar{a} + a)\bar{b}c$$

Podle pravidla 5 byl zdvojen střední člen, podle pravidla 4 byly z dvojic členů vytčeny společné proměnné. V dalším postupu aplikujeme na závorky pravidlo 8 a vytkneme společnou proměnnou podle pravidla 4.

$$y = \bar{a}\bar{b} + \bar{b}c = \bar{b}(\bar{a} + c) \qquad (3)$$

Obdobně můžeme upravit vztah (2). Podle pravidla 7 celý výraz dvakrát negujeme a aplikujeme pravidlo 9:

$$y = \overline{\overline{y}} = \overline{\overline{(\bar{a} + b + c)} + \overline{(a + \bar{b} + c)} + \overline{(\bar{a} + \bar{b} + c)} + \overline{(a + \bar{b} + \bar{c})} + \overline{(\bar{a} + \bar{b} + \bar{c})}}$$

Aplikací pravidla 9 na negované závorky, v dalším kroku vytknutím podle pravidla 4 a nakonec aplikací pravidla 8 obdržíme:

$$y = \overline{a\bar{b}\bar{c} + \bar{a}\bar{b}c + ab\bar{c} + \bar{a}bc + abc} = \overline{a\bar{b}\bar{c} + (\bar{a} + a)b\bar{c} + (\bar{a} + a)bc} = \overline{a\bar{b}\bar{c} + b\bar{c} + bc}$$

Zdvojením středního členu podle pravidla 5, vytknutím společného členu ve dvojicích podle 4, aplikací 6 na prvou a 8 na druhou závorku a po roznásobení prvé závorky obdržíme:

$$y = \overline{a\bar{b}\bar{c} + b\bar{c} + b\bar{c} + bc} = \overline{(a\bar{b} + b)\bar{c} + b(\bar{c} + c)} = \overline{(a + b)\bar{c} + b} = \overline{a\bar{c} + b\bar{c} + b}$$

Vytknutím *b*, aplikací 1 na závorku a konečnou aplikací 9 na celý výraz a dále na prvý člen získáme výsledný tvar shodný s tvarem (2):

$$y = \overline{a\bar{c} + b(\bar{c} + 1)} = \overline{a\bar{c} + b} = \overline{a\bar{c}} \cdot \bar{b} = (\bar{a} + c)\bar{b} \qquad (4)$$

Při úpravě výsledného výrazu na vyjádření negovaným logickým součinem, neboli Shefferovou funkcí se obvykle postupuje tak, že se výraz dvakrát neguje a aplikuje se pravidlo 9. Tím se prakticky dostaneme k předposlednímu vyjádření výsledků ve vztahu (4).

Minimalizace pomocí map

Velmi přehlednou minimalizaci umožňují mapy. Z řady různých možných druhů map patří mezi nejznámější ***mapa Karnaughova*** a ***mapa Svobodova***. Počet polí u obou map je stejný, neboť odpovídá počtu možných kombinací na vstupu.

Výhodou Karnaughovy mapy je, že sousední prvky jsou umístěny vedle sebe. Nevýhoda mapy spočívá v nepohodlném zápisu prvků z tabulky.

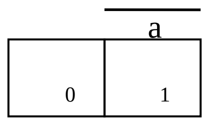
Svobodova mapa umožňuje snadnější a přehlednější zápis prvků z tabulky, ale proměnné jsou promíchány.

Mapa je grafickým znázorněním Booleovy funkce, která vyjadřuje vztah závislé a nezávislé proměnné. V mapě vyhledáváme sousedící dvojice, čtveřice, osmice, ... logických jedniček. Protože mapa je uspořádána tak, že pro přechod z jednoho pole označeného jedničkou na sousedící pole označené rovněž jedničkou se musí změnit hodnota jedné vstupní proměnné, pak z toho plyne, že v tomto případě dosahuje logická funkce hodnoty 1 pro oba stavy této vstupní proměnné, tedy na této vstupní proměnné nezávisí. Do výsledného výrazu pak tuto proměnnou nezapisujeme. Jedna hodnota logické 1 může být užita vícekrát. Výsledný výraz pak určíme tak, že pro společnou skupinu logických jedniček zapíšeme součin pouze těch vstupních veličin, které se v označené skupině nemění, a to podle pravidel pro zápis úplná disjunktní normální formy (ÚDNF). Užitím map můžeme velmi rychle dojít k minimalizovanému výrazu.

Základní nevýhodou map je to, že jimi můžeme řešit úlohy nejvýše s 5 - 6 proměnnými. Byly sice sestrojeny mapy s větším počtem proměnných a byly pro ně určeny různé způsoby, ale zpracování výrazů, které je nutné k dosažení výsledků je u nich poněkud složité. Proto se vesměs omezujeme na mapy pro 4 proměnné.

Karnaughova mapa pro:

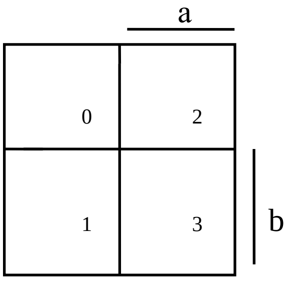
1 proměnnou



pod pruhem je hodnota logická 1, s = index pole

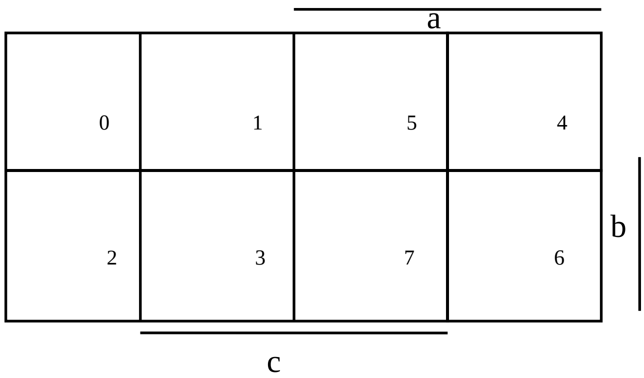
s	a	f
0	0	$\bar{a}$
1	1	a

2 proměnné

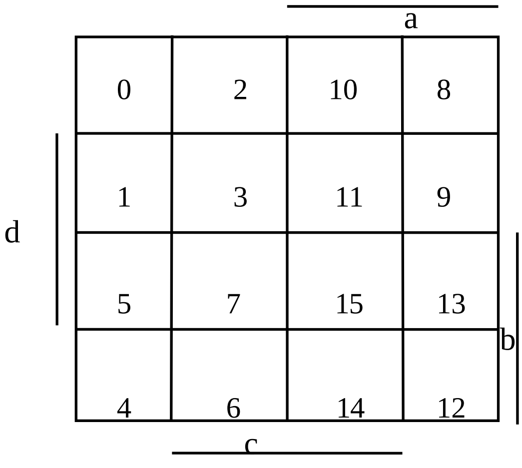


s	a	b	f
0	0	0	$\bar{a} . \bar{b}$
1	0	1	$\bar{a} . b$
2	1	0	$a . \bar{b}$
3	1	1	$a . b$

3 proměnné



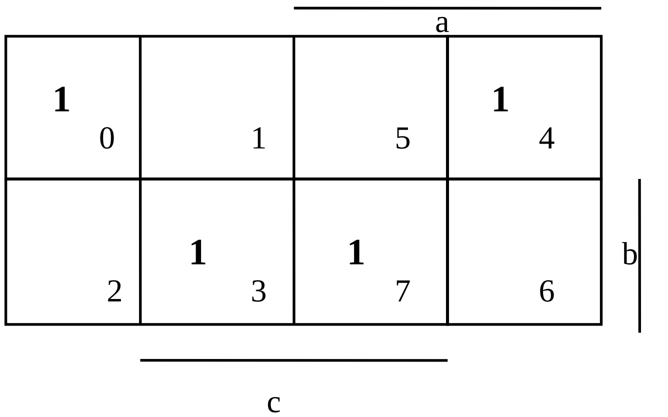
4 proměnné



Víme, že každé políčko mapy nám odpovídá jednomu stavu z dané pravdivostní tabulky a že tedy jeho poloha v mapě jednoznačně určuje základní konjunkci (minterm) dané funkce. K tomu abychom mohli zobrazit danou funkci, musíme do těchto políček v mapě, které odpovídají jedničkovým mintermům dané funkce dosadit 1.

Mějme funkci o třech vstupních proměnných *a*, *b*, *c* zadanou tabulkou.

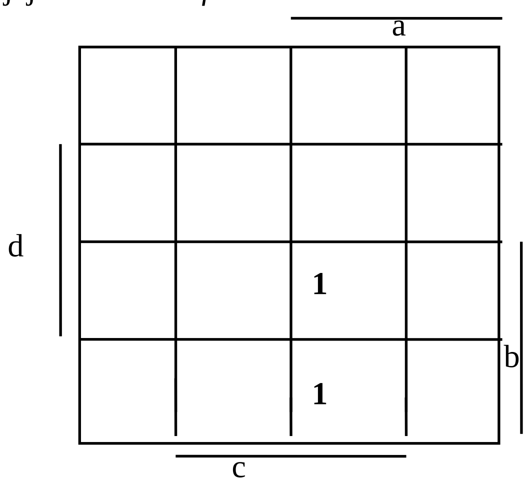
a	b	c	f
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1



Funkce je úplně určena´, je-li určena hodnota v každém políčku mapy (pro přehlednost zapisujeme do mapy pouze 1 a předpokládáme, že prázdná políčka jsou 0). Úplná disjunktní normální forma (ÚDNF) pak bude mít tvar:

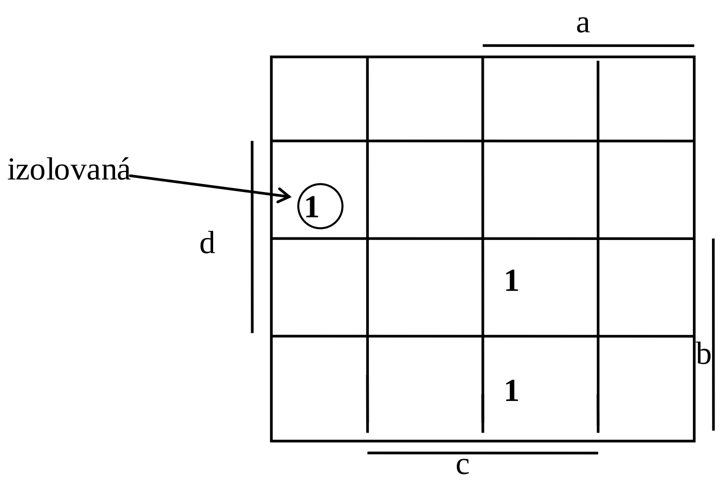
$$f = \bar{a}\bar{b}\bar{c} + \bar{a}bc + a\bar{b}\bar{c} + abc$$

Mějme funkci, jejíž ÚDNF $f = abcd + abcd$



Vidíme, že uplatněním zákonů logické algebry jsme si danou funkci zjednodušili, protože jak se ukázalo proměnná *d* je nepodstatná a nevyskytuje se v konečném výrazu.

Je to proto, že mintermy dané funkce jsou políčky sousedními a liší se v jedné proměnné. Konjunkci o 4 proměnných nazýváme konjunkci 4-tého řádu a konjunkci nazýváme ***izolovanou***, pokud neexistuje žádná sousední konjunkce.



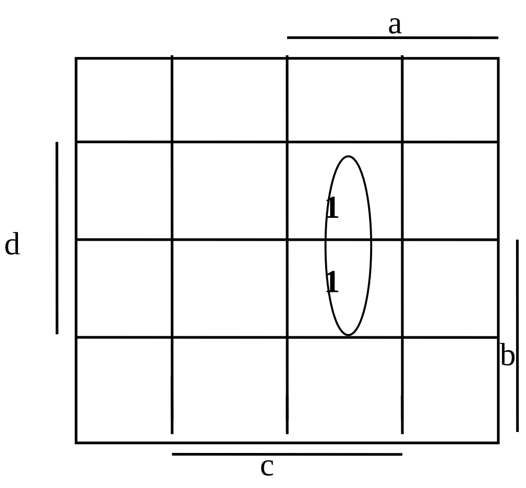
Závěry:

1. Všechna políčka mapy jsou přiřazena různým kombinacím hodnot všech vstupních proměnných.
2. Sousední políčka:

   - mají jednu stranu společnou

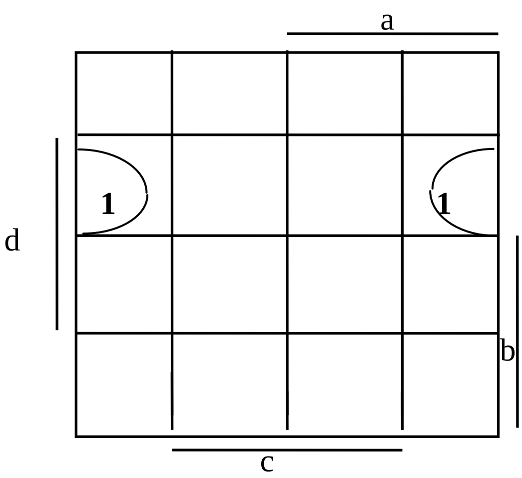
   - přísluší kombinacím hodnot proměnných, z nichž jedna má různé hodnoty.
3. Liší-li se dva členy v hodnotě jedné proměnné, můžeme je spolu zkrátit na jeden člen podle věty $ab + ab = a$, kde a může zastupovat i složitější výraz. Z toho vyplývá pravidlo, kterým se řídíme při čtení sousedních políček:

   Sousední jedničková políčka můžeme číst z mapy jako jeden člen, který obsahuje pouze proměnné, jejichž hodnota je pro uvažovaná políčka stejná. Čteme-li funkci z mapy, můžeme si pomoci uzavřením sousedních jedniček, které spolu sousedí do tzv. ***smyčky***.



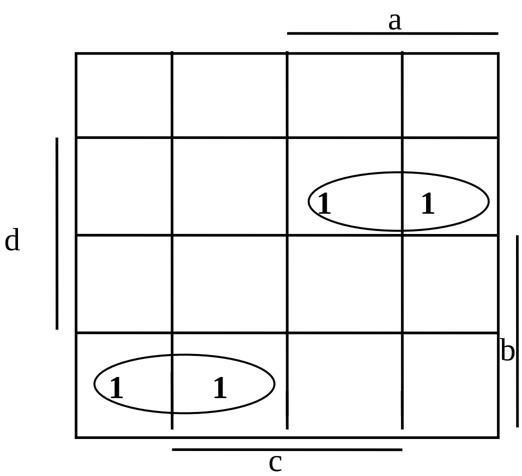
f = acd

Obsahuje-li smyčka dvě jedničky, nazýváme ji ***dvojsmyčkou***. Každá dvojsmyčka obsahující dvě sousední jedničky nám vylučuje jednu z logických proměnných.



Zde máme další dvojsmyčku a vidíme, že na hodnotu výsledné funkce zde nemá vliv proměnná *a*. Provedeme-li rozbor všech možných dvojic jedniček zaznamenaných v mapě, zjistíme že, sousední políčka jsou taková, která leží vedle sebe ve stejném sloupci nebo řádku nebo ta políčka, která leží na konci téhož řádku nebo téhož sloupce.

Všimněme si, že pokud čteme funkci z mapy, má výsledná funkce o proměnnou méně než mapa. Je-li v mapě několik dvojic jedniček uzavřených dvojsmyčkou, čteme je postupně nezávisle na sobě.

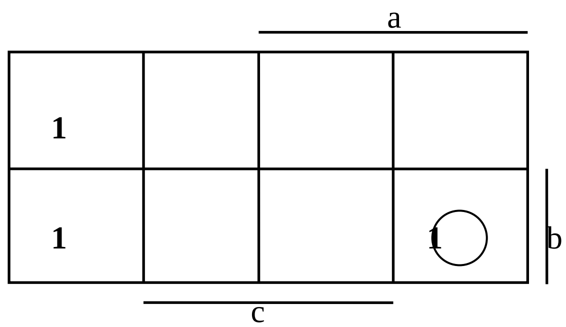


V tomto případě jedničky v mapě můžeme uzavřít do dvou dvojsmyček a výsledná funkce má tvar $f = abd + abd$.

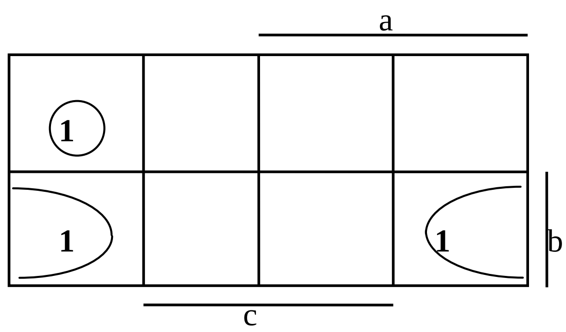
Ukažme si, že se obecně může vyskytnout několik řešení při čtení z výsledné mapy. Máme zde 3 mapy, které nám zobrazují jednu a tutéž funkci.

V prvním případě bude funkce zapsána: $f = \bar{a}\bar{c} + a\bar{b}c$,
ve druhém případě: $f = \bar{a}\bar{b}\bar{c} + \bar{b}c$
a ve třetím případě: $f = \bar{a}\bar{c} + b\bar{c}$

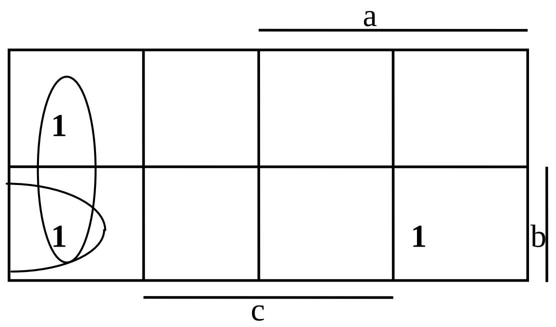
Z uvedeného vyplývá, že jedna jednička se může uzavřít do libovolného počtu dvojsmyček, přičemž se snažíme, aby výsledný výraz pro funkci byl co nejjednodušší.



$f = \bar{a}\bar{c} + ab\bar{c}$



$f = \bar{a}\bar{b}\bar{c} + b\bar{c}$



$f = \bar{a}\bar{c} + b\bar{c}$

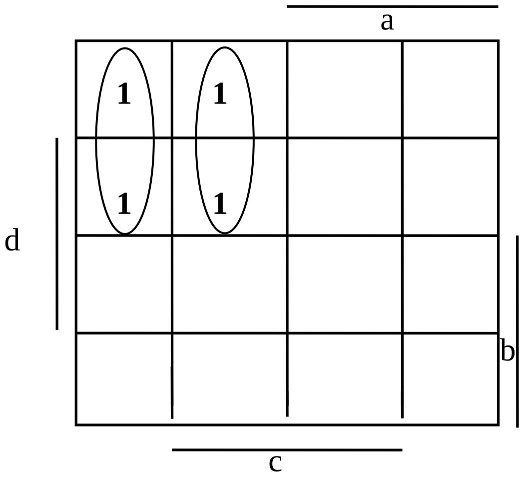
Mějme funkci zadanou mapou.

Tuto funkci může vyjádřit dvěmi dvojsmyčkami

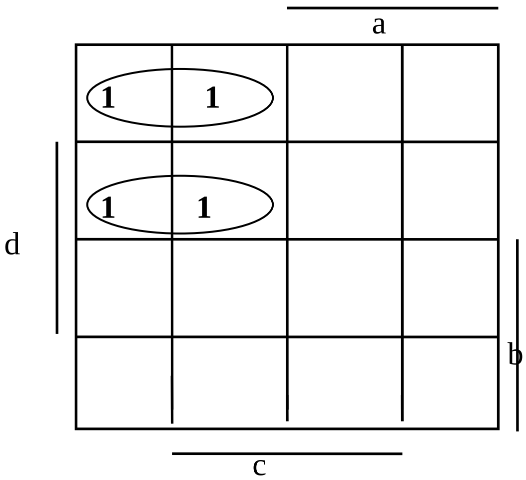
$f = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c = \bar{a}\bar{b}.\,(\,\bar{c} + c\,) = \bar{a}\bar{b}$

$f = \bar{a}\bar{b}\bar{d} + \bar{a}\bar{b}d = \bar{a}\bar{b}\,.\,(\,\bar{d} + d\,) = \bar{a}\bar{b}$

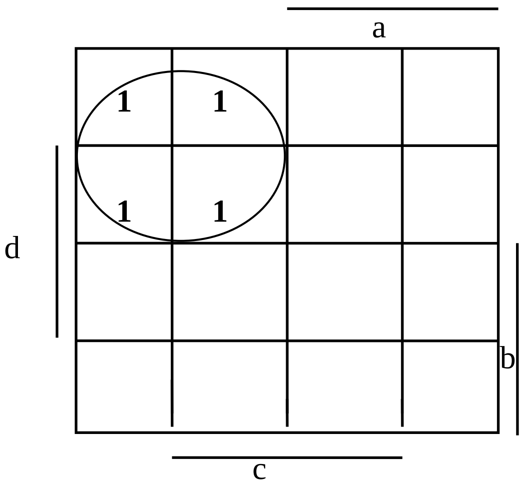
V mapě můžeme uzavřít tzv. ***čtyřsmyčku*** která nám umožní vyloučit další proměnnou. Vytvoření čtyřsmyčky eliminuje vliv další proměnné.



$f = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}c$

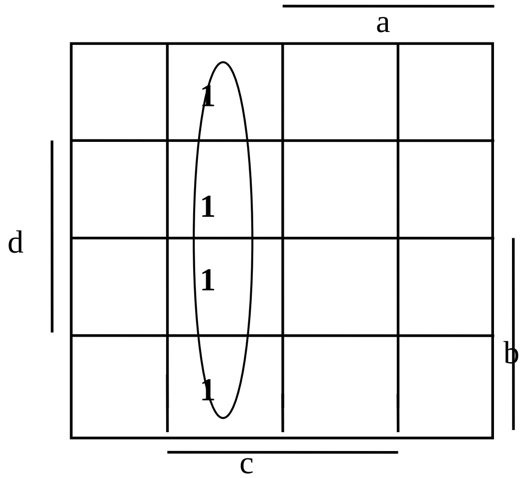


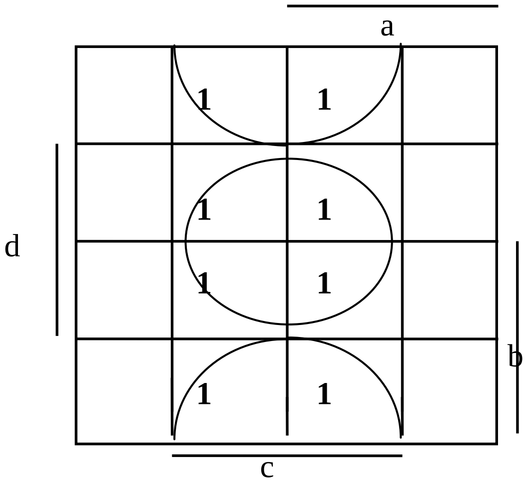
$f = \bar{a}\bar{b}\bar{d} + \bar{a}\bar{b}d$

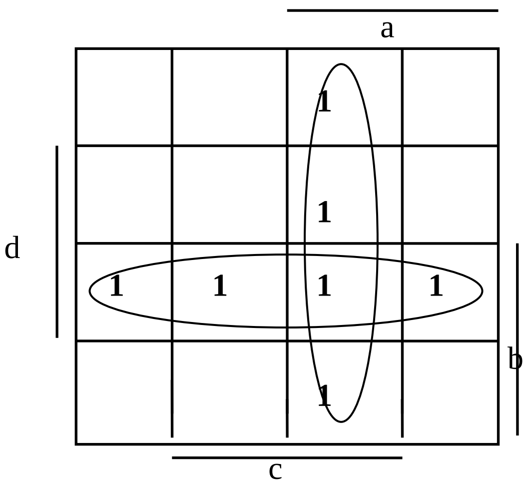


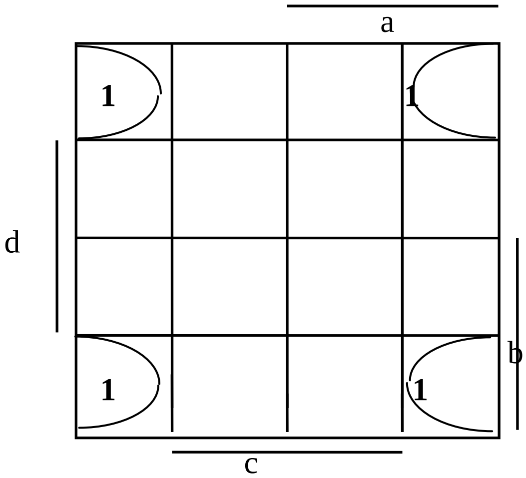
$f = \bar{a}\bar{b}$

Čtyřsmyčky můžeme uzavírat následujícím způsobem.









Dvě sousední dvojsmyčky je tedy možno spojit v jednu čtyřsmyčku. Podobně dvě čtyřsmyčky můžeme spojit v jednu osmismyčku.

Pro mapy o větším počtu proměnných bychom spojováním mohli vytvářet smyčky dalších vyšších řádů.

Účelem mapy je minimalizace výsledného výrazu. Větší smyčky vedou k jednoduššímu výrazu, který je vytvářen menším počtem proměnných. Určujeme-li smyčky, snažíme se dosáhnout co nejmenšího počtu co největších smyček.

Při minimalizaci pomocí Karnaughových map postupujeme následovně:

- Probíráme postupně jednotlivá políčka jedno po druhém a všímáme si, můžeme-li je uzavřít jedinou smyčkou nebo několika různými smyčkami.
- Můžeme-li je uzavřít jedinou smyčkou jediným způsobem, zakreslíme smyčku, zapíšeme výraz a pokračujeme.
- Můžeme-li políčko uzavřít do několika smyček, políčko pomineme. Nebereme v úvahu, zasahují-li smyčky navzájem do sebe.
- Zbývající políčka uzavřeme do co nejmenšího počtu co největších smyček.

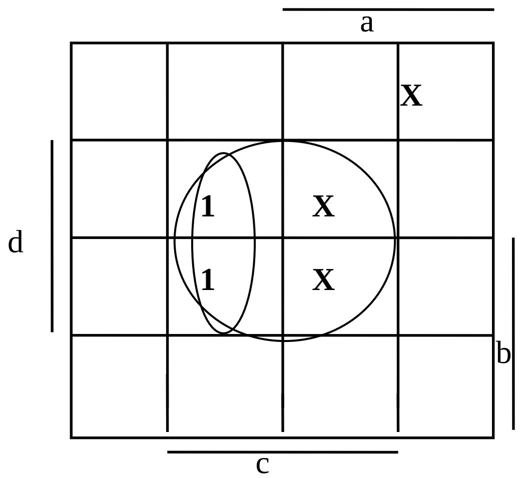
Využití neurčitých stavů

Jak již bylo uvedeno, některé stavy logických proměnných nemusí být zadány, jsou to tzv. ***neurčité stavy***. Tyto neurčité stavy můžeme využívat velmi jednoduše a mohou být pomůckou při minimalizaci.

Jelikož neurčité stavy můžeme považovat buď za 0 nebo za 1, volíme jejich hodnoty tak, abychom vytvořili co největší smyčku, protože čím větší smyčka, tím jednodušší je výsledný výraz.

Můžeme formulovat obecné pravidlo pro práci s neurčitými stavy:

Každou jedničku se v mapě snažíme uzavřít do největší možné smyčky, která musí obsahovat buď další jedničky, nebo neurčité stavy. Vybereme nejmenší počet smyček, které obsahují všechny jedničky, neurčité hodnoty, pokud jsou uvnitř smyček, pokládáme za jedničky. Ostatní považujeme za nuly. Smyčky, které obsahují pouze neurčité hodnoty, nezahrnujeme do výsledného vztahu.



V uvedeném případě je výhodné uzavřít čtyřsmyčku. Třetí neurčitý výraz budeme považovat za 0, protože nezvyšuje hodnotu žádné smyčky.

Výsledná funkce

- v případě nepoužití neurčitých stavů: $f = \bar{a}cd$
- v případě využití neurčitých stavů: $f = cd$

∑ Shrnutí pojmů

Logická proměnná, logická funkce, základní logické funkce, Operace s logickými proměnnými, realizace logických funkcí, vyjádření logických funkcí, minimalizace logických výrazů, využití neurčitých stavů.

? Otázky

1. Popište základní logické funkce.
2. Uveďte a vysvětlete základní zákony Booleovy algebry.
3. Popište způsoby realizace a minimalizace logických funkcí.
4. Co jsou to neurčuj stavy a k čemu

Další zdroje

Tomis, L. - Němec, F. - Balcová, J. : *Základy teorie systémů*, skripta VŠB, Ostrava, 1989

Vrožina, M. - Koběrský, J. : *Základy automatizace technologických procesů*, učební texty dálkového studia FMMI, VŠB - TU, Ostrava, 1998
A : *Systémové inženýrství*

Pitra, Z.: *Teorie systémů*, MŠMT ČSR, Praha, 1989

Štecha, J.: *Obecná teorie systémů*, skripta ČVUT, Praha, 1981

Soukup, J.: *Identifikace systémů*, skripta ČVUT, Praha 1986

Klír, J., Seidl, K.: *Syntéza logických obvodů*, SNTL, Praha, 1966

Vítečková, M.: *Matematické metody v řízení, L- a Z- transformace*, VŠB-TU Ostrava, 1998