

VYSOKÁ ŠKOLA STROJNÍ A TEXTILNÍ V LIBERCI

Fakulta strojní

Katedra výrobních systémů

Školní rok: 1991 - 92

ZADÁNÍ DIPLOMOVÉ PRÁCE

pro Jiřího Č E C H A

obor (23-19-8) výrobní systémy s průmyslovými roboty
a manipulátory

Vedoucí katedry Vám ve smyslu zákona č. 172/1990 Sb. o vysokých školách určuje tuto diplomovou práci:

Název tématu: Automatizovaný návrh mechanické úchopné hlavice

Zásady pro vypracování:

1. Skladba mechanických úchopných hlavic, problematika jejich návrhu (objekt manipulace, úchopná síla, kinematika transformačního bloku, pohon atd) a možnosti automatizovaného návrhu - úvod do problému.
2. Vypracování programu pro automatizovaný návrh vybraných kinematických struktur transformačního bloku (kinematické veličiny, silové poměry, pevnostní řešení vybraných mechanických částí ap).
3. Programový produkt, popis systému, komentovaný výpis programu, návod k použití.
4. Příklad automatizovaného návrhu mechanické úchopné hlavice.

VYSOKÁ ŠKOLA STROJNÍ A TEXTILNÍ
Odbor knihovny
STUDENTSKÁ 5
602 00 LIBEREC

KVS/YSnem

V 19/92

Rozsah grafických prací: podle potřeby
Rozsah průvodní zprávy: cca 50 stran textu vč. obrázků
Seznam odborné literatury:
Volmer, J.: Industrieroboter VEB Verlag Technik, Berlin 1983
Matička, R.-Talácko, J.: Konstrukce průmyslových robotů. ČVUT,
Praha 1989
Technická dokumentace vývojového systému AUTOCAD a AUTOLISP.
AUTODESK, 1991
Technická dokumentace a návod k použití překladače TURBC++.

Vedoucí diplomové práce: Doc. Ing. Josef Cerha, CSc.
Konzultant:

Zadání diplomové práce: 31.10.1991
Termín odevzdání diplomové práce: 29.5.1992



Josef Cerha
Doc. Ing. Josef Cerha, CSc.
Vedoucí katedry

J. Exner
Prof. Ing. J. Exner, CSc.
Děkan

Vysoká škola strojní a textilní v Liberci
fakulta strojní

Obor: 23-19-8
Výrobní systémy s průmyslovými roboty a manipulátory

Katedra výrobních systémů

Automatizovaný návrh mechanické úchopné hlavice

KVS – VSPR – SF – 19

Jméno a příjmení autora: Jiří Čech

Vedoucí diplomové práce: Doc. Ing. Josef Cerha, CSc.

Rozsah práce a příloh:

Počet stran: 62

Počet obrázků: 6

Počet příloh: 14

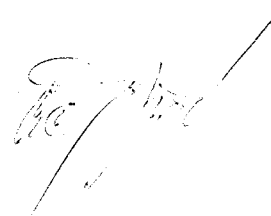
UNIVERZITNÍ KNIHOVNA
TECHNICKÉ UNIVERZITY V LIBERCI



3146076581

Místopřísežně prohlašuji, že jsem diplomovou práci vypracoval samostatně
s použitím uvedené literatury.

V Liberci 28.5.1992

A handwritten signature in dark ink, appearing to be 'F. J. ...', written over a horizontal line.

Děkuji touto cestou svému vedoucímu diplomové práce Doc.Ing.Josefu Cerehovi,Csc. za metodickou pomoc a poskytování informací při vypracování diplomové práce

Jiří Čech

Seznam použitých zkratek:

ISO	International Standard Organization
ANSI	American National Standard Institution
CAD	Computer Aided Design
CAM	Computer Aided Manufacturing
CIM	Computer Integrated Manufacturing
GKS	Graphical Kernel System
PHIGS	Programmer's Hierarchical Interactive Graphics System
CSG	Constructive Solid Geometry
XBF	Experimental Boundary File
B-rep	Boundary representation
CAX	souhrnné označení systémů Computer Aided
VDA-IS	Verband der Automobilindustrie IGES Subset
VDA-FS	Verband der Automobilindustrie Flaschenschnittstelle
PDES	Product Data Exchange Specification
STEP	The Standard for the Exchange of Product Model Data
PDDI	Product Data Definition Interchange
DXF	Drawing Interchange File
ADS	AutoCAD development system
MÚH	Mechanická úchopná hlavice
PRaM	Průmyslové roboty a manipulátory

Následující ochranné známky jsou registrovány jako U.S. patenty a ochranné známky Office Autodesk Inc: AutoCAD, Autodesk, AutoLISP, AutoCAD Development System, DXF.

Obsah

Seznam použitých zkratk	5
1 Úvod	8
2 Skladba mechanických úchopných hlavíc	10
3 Automatizovaný návrh MÚH	17
3.1 Možnosti automatizovaného návrhu	17
3.2 Rozbor prostředků automatizovaného návrhu	19
4 Program pro automatizovaný návrh MÚH	26
4.1 Výběr prostředku pro automatizovaný návrh	26
4.2 Požadavky na program	27
4.3 Popis programu	28
4.3.1 Rámcový popis programu	28
4.3.2 Stručný popis globálních proměnných	30
4.3.3 Stručný popis nově definovaných funkcí	33
4.3.4 Stručný popis obsahu zdrojových souborů	35
4.3.5 Seznam dalších definovaných prvků	38
4.3.6 Další rozšíření programu	42
4.4 Návod k použití programu	43
4.4.1 Postup návrhu mechanické úchopné hlavice	43
4.4.2 Postup při kontrole již navržené hlavice	50
4.4.3 Podrobný popis menu HLAVICE	52

5	Příklad automatizovaného návrhu MÚH	59
5.1	Zadání požadované hlavice	59
5.2	Příklad řešení automatizovaného návrhu hlavice	59
6	Závěr	60
A	Seznam použité literatury	61
B	Seznam příloh	62

1 Úvod

V posledních několika letech se ve strojírenství stále více setkáváme s prudkým rozvojem zavádění automatizovaných výrob, způsobeným především revolučním vývojem elektroniky. Tento vývoj je markantní nejen v oblasti nových technologií, nových možností využití, ale bezesporu i cen jednotlivých komponentů. Nasazení cenově přístupné, současně však spolehlivé řídicí elektroniky umožňuje vyrábět univerzální stroje, které lze na jedné straně vyrábět ve velkých sériích a na druhé straně využívat pro různé činnosti doposud prováděné více speciálními stroji. Tímto jako další podstatný faktor přichází v úvahu větší pružnost výrobních celků sestavených z těchto strojů.

Nejedná se však pouze o výrobní stroje, ale i o dopravní a manipulační zařízení. Právě u průmyslových robotů a manipulátorů (PRaM) je vývoj v této oblasti nejmarkantnější. V minulosti jsme se mohli s PRaM setkat také, ale jednalo se právě z důvodů nespolehlivých a vývojově nižších řídicích systémů o zařízení drahá a pochopitelně nespolehlivá, vyráběná většinou osamocenými výrobci nejčastěji pro vlastní potřebu a především do provozů pro člověka nebezpečných (výpary, záření, hluk atd.).

Nyní se však již výrobou PRaM zabývají specializované firmy produkující levná a vysoce spolehlivá zařízení. Tito výrobci nabízejí široké spektrum možných konfigurací PRaM podle přání zákazníků.

Zákazník většinou zakoupí určitý typ průmyslového robotu a zpravidla přímo s ním řídicí systém. V případě, že se jedná o standardní využití, je součástí dodávky i patřičná výstupní hlavice. Tady se však zpravidla dostaví problémy. Nabídka výstupních hlavic nemůže být ani při maximální snaze výrobců úplná, aby uspokojila potřeby všech zákazníků. Zákazník tedy stojí zpravidla před problémem návrhu a vlastní výroby požadované výstupní hlavice.

Výstupní hlavice můžeme rozdělit například podle způsobu použití na:

- úchopné

- technologické
- kontrolní
- speciální

Pokud se budeme dále zabývat pouze úchopnými hlavicemi, lze provést další dělení na:

- hlavice s magnetickými úchopnými prvky
- hlavice s podtlakovými úchopnými prvky
- hlavice s mechanickými úchopnými prvky

Pokud bychom byli nuceni vyrábět výstupní úchopnou hlavici, bude se jednat ve většině případů o hlavici s mechanickými úchopnými prvky, protože nabídka předcházejících dvou typů hlavic bývá zpravidla dostačující. V opačném případě se však jedná o jednodušší návrh s experimentálním ověřením.

Jiný případ je právě hlavice s mechanickými úchopnými prvky. Konstruktor se při návrhu mechanické úchopné hlavičky zpravidla snaží o co možná nejjednodušší řešení. Podle typů úchopných prvků můžeme hlavice rozdělit na:

- hlavice s aktivními úchopnými prvky
- hlavice s pasivními úchopnými prvky,

Pokud je alespoň jeden z úchopných prvků aktivní, jedná se o hlavici s aktivními úchopnými prvky. Rozdíl mezi pasivním a aktivním úchopným prvkem spočívá v tom, že u aktivního úchopného prvku můžeme ovládat velikost úchopné síly, kdežto u pasivního nikoliv. U těchto prvků se však často využívají pružné nebo odpružené úchopné prvky.

2 Skladba mechanických úchopných hlavíc

Mechanické úchopné hlavice reprezentují výstupní funkční členy otevřených kinematických struktur průmyslových robotů. Většina mechanických úchopných hlavíc má tyto společné znaky:

- pohyb aktivních prvků z uvolněné polohy do polohy bezprostředního mechanického kontaktu s povrchem uchopovaného objektu a následné vyvození uchopovacích sil
- držení objektu koncovými uchopnými prvky v průběhu programově vymezeného prostorového pohybu průmyslového robotu v rámci manipulačního cyklu
- uvolnění uchopovaných prvků od povrchu objektu, přičemž rozhodující fází pracovního režimu je ta, ve které chapadlo a uchopený objekt tvoří přes vzájemnou silově-kontaktní vazbu integrovaný celek
- tyto činnosti jsou programově řízeny a zpětně kontrolovány.

Na mechanické úchopné hlavice je kladena řada požadavků souvisejících s jejich nasazením v reálném provozu:

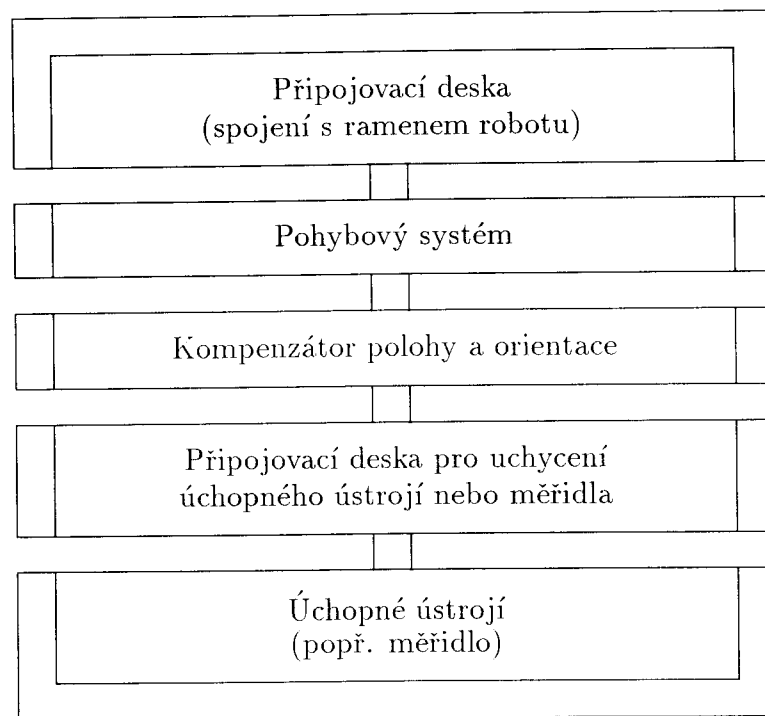
- zabezpečit stabilní opakovatelnou polohu uchopených objektů v rozsahu dovolené nepřesnosti uchopení
- nezpůsobit v místě mechanického kontaktu nežádoucí deformace, destrukce nebo jiné nežádoucí efekty
- zabezpečit spolehlivou funkci v průběhu projektované životnosti při předepsaných opravách a údržbě
- umožnit optimální využití robotu, pro který jsou určeny
- vlastní konstrukce chapadel by měla být podle možnosti v optimálních rozměrových dimenzích, tvaru a hmotnosti vzhledem k uchopovaným objektům i příslušnému robotu

- mechanika mechanické úchopné hlavice by měla dovolovat jednoduché, spolehlivé a přesné seřízení, výměnu i případné opravy v provozních podmínkách
- z hlediska designu by měla být mechanická úchopná hlavice kompaktní s robotem.
- výroba mechanické úchopné hlavice by měla být co do nákladů úměrná účelu a funkci
- při výpadku energie je potřeba zajistit bezpečné udržení objektu v chapadle
- zajistit bezpečné uchopení i při nejnepríznivějších podmínkách (vibrace, zvýšená teplota atd.)
- zajištění informací o uchopení objektu pro řídicí systém
- zabezpečení požadované rychlosti uchopení
- zabezpečení kompenzace odchylek ve vychystání objektu, popřípadě umožnit měření a programovou korekci odchylky
- umožnit stavebnicovost a modularitu jak v konstrukci hlavice, tak v jejím použití
- umožnit rychlý přechod na nový objekt manipulace

V převážné většině případů však nelze vyhovět všem požadavkům zcela a proto je na konstruktérovi, aby si předem určil patřičné priority jednotlivých požadavků.

Pro větší pružnost výroby se zpravidla konstruktér snaží o větší rozsah uchopovaných průměrů.

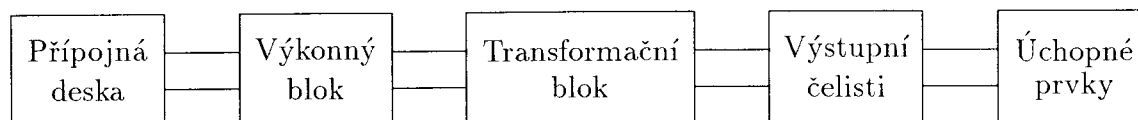
Struktura výstupní hlavice může být následující:



Obr. 1:

Skladba mechanické úchopné hlavice

Mechanická úchopná hlavice se skládá z těchto základních stavebních celků:



Obr. 2:

Přípojná deska

V praxi se vyskytuje v podstatě dvojí řešení realizace přípojně desky. Buď je přípojná deska přímo součástí rámu PRaM, anebo řešením na první pohled složitějším a náročnějším je využití některého ze systémů výměnných přípojných desek. Jako odměna za složitější konstrukci zde však přichází větší flexibilita takto vzniklých výměnných úchopných hlavice.

Pohonný systém hlavice

Ve vztahu k úchopné hlavici lze rozdělit pohonný systém hlavice na pohon

- oddělený
- integrovaný přímo do hlavice

Z hlediska druhů pohonů se jedná o pohony

- hydraulické
- pneumatické
- elektrické
- elektromagnetické

a podle vyvozovaného pohybu

- lineární
- rotační

Budeme se dále zabývat pouze častěji používanými řešeními. Jedná se především o přímočaré hydraulické a pneumatické motory. Pro jejich využití mluví především nenáročná konstrukce nevyžadující složitou údržbu, ani speciální nároky na pracovní prostředí (přetížení, prach, vlhkost, nevýbušnost atd.). V neposlední řadě jsou pro tuto kategorii pohonů charakteristické malé rozměry v poměru k vyvozovaným silám a jejich snadná regulace pomocí redukčních ventilů. Nevýhodou je zde však nutnost externího zdroje tlakové energie a poněkud neskladného přívodu. Z hlediska zpětného pohybu můžeme zavést dělení na motory

- dvoučinné
- jednočinné s pružinou

Další kategorií pohonů jsou elektromotory. Použití přímočarých motorů není dosud příliš časté z důvodu technické náročnosti. Jiná situace je u točivých elektromotorů, i když proti nim mluví potřeba převodovky z důvodu relativně vysokých otáček. Nezanedbatelnou nevýhodou těchto pohonů je i relativně malý vyvozovaný kroutící moment a relativně vysoké nároky na pracovní podmínky (přetížení, vlhkost atd.). Nespornou výhodou elektrických pohonů je však velmi jednoduché řízení bez potřeby dalších mezičlánků (energetických transformací) přímo z řídicího systému.

Transformační blok

Transformační blok plní tyto funkce:

- změnu druhu pohybu
- změnu smyslu pohybu
- změnu velikosti úchopné síly
- změnu velikosti rychlosti pohybu
- společné ovládání více čelistí

Transformační bloky mohou realizovat změnu charakteru a smyslu pohybu. Lze je i řetězit za sebou ve snaze o dosažení požadované transformace s ohledem na co nejmenší zastavěný prostor a z toho vyplývající výslednou hmotnost úchopné hlavice s tímto transformačním blokem. Přes všechny výhody, které nám z mechanického hlediska poskytují transformační bloky však platí zásada, že pokud je to jen trochu možné, transformační blok nepoužijeme, a to nejen z důvodů již uvedené hmotnosti, ale i ztrát, ke kterým zákonitě v každém transformačním bloku dochází.

Čelisti

Čelisti mohou vykonávat pohyb:

- rotační
- posuvný
- kombinovaný (paralelogram)

Nejčastěji používané jsou čelisti rotační díky jednoduché konstrukci a především poměrně malému zastavěnému prostoru, který bývá u úchopných hlavice pro univerzální použití velmi důležitý. Hlavice je tím pádem elegantnější, lehčí a má lepší schopnosti pro práci ve stísněných prostorech.

Opakem již zmíněného je konstrukce posuvných čelistí, kde navíc přistupuje potřeba přesnější výroby a vymezení vůlí. Po tomto řešení však musíme sáhnout v případě, že požadujeme stálý střed uchopovaných součástí nezávislý na uchopovaném průměru bez použití speciálních úchopných prvků. Kompromisem mezi těmito dvěma řešeními je provedení čelistí s paralelogramem.

Úchopné prvky

Úchopné prvky jsou posledním článkem řetězce mechanické úchopné hlavice uskutečňujícím konečné uchopení manipulovaného objektu. Jsou na ně proto kladeny

poměrně vysoké nároky. Budeme se dále věnovat pouze oblasti uchopování rotačních objektů. Pomineme-li úchopné prvky, které jsou neoddělitelnou částí čelistí, můžeme provést rozdělení základních druhů úchopných prvků.

- prizmatické pevné
- prizmatické uložené na čepu
- ploché
- speciální s přesně definovanými plochami (tyto čelisti umožňují centrické uchopení objektu i u rotačně uložených čelistí)

Podle flexibility můžeme rozdělit úchopné prvky také na

- neoddělitelné
- výměnné

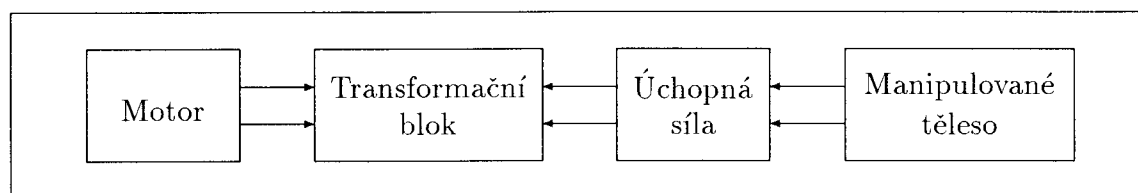
Nejpoužívanějším typem úchopných prvků jsou z důvodu velmi jednoduché konstrukce a poměrně stabilního uchopení pevně uložené prizmatické prvky. Pro použití rotačně uložených čelistí pro různé uchopované průměry při stejném středu uchopovaného objektu jsou určeny speciální úchopné prvky s přesně definovanými kontaktními plochami.

Pro použití neoddělitelných úchopných prvků mluví především výsledná tuhost uchopení, jednoduchost konstrukce a menší hmotnost. Naopak bez výměnných prvků se neobejdeme, pokud konstruujeme mechanickou úchopnou hlavici pro širší použití.

3 Automatizovaný návrh MÚH

3.1 Možnosti automatizovaného návrhu

Možnosti automatizovaného návrhu mechanických úchopných hlavíc můžeme rozdělit do několika úseků, které na sebe velmi úzce navazují.



Obr. 3:

Prvním úsekem při návrhu mechanické úchopné hlavice je výpočet potřebné minimální úchopné síly na základě známých charakteristických parametrů manipulovaného objektu a dynamických schopností použitého robotu. Zde nalezne uplatnění automatizace v první fázi ve výpočtu hmotnosti a umístění těžiště pro výběr optimálního středu uchopení, což je při ručním výpočtu složitých součástí činnost zbytečně pracná a rutinní. Z tohoto důvodu se nabízí možnost nasazení jednoduchého programu.

Následuje výpočet potřebné úchopné síly, který se mění podle typu použitých úchopných prvků. Ani zde se opět v žádném případě nejedná o tvůrčí činnost, a proto je i zde prostor pro jednoduché programy provádějící tyto výpočty.

V případě návrhu motoru máme zpravidla buď určené přípojně rozměry, maximální průměr či velikost zastavěného prostoru, nebo přímo jeden či více konkrétních typizovaných motorů. Tím pádem nemáme mnoho možností jaký motor navrhnout, přičemž ani výpočet síly na motoru není nikterak složitý.

Všechny doposud uvedené úseky návrhu by bylo možné automatizovat například pomocí programového balíku, který by obsahoval potřebné výpočtové vztahy

pro všechny častěji používané druhy motorů, úchopných prvků a některých typických objektů manipulace, který by sloužil konstruktérovi jako pomůcka vedle kalkulačky.

Jiná situace nastává však v posledním úseku návrhu mechanické úchopné hlavice, kterým je transformační blok. Je tomu hlavně z toho důvodu, že většinou všechny ostatní prvky máme již pevně dány (objekt, úchopné prvky, motor) a transformační blok je právě tím úsekem, který by měl zajistit potřebný převod mezi nimi. Můžeme zpravidla použít veliké množství různých struktur transformačních bloků a kromě toho v rámci každé struktury velké množství rozměrových, poměrových a prostorových uspořádání.

Při návrhu transformačního bloku sledujeme mnoho hledisek jakými jsou například dodržení potřebné minimální úchopné síly podle vypočtené maximální síly na motoru, rozsah uchopovaných průměrů pro daný rozsah zdvihu motoru, odchylka středu uchopení a další. Musíme však v rámci návrhu použít kompromis mezi těmito a dalšími požadavky. Jedná se o návrh kinematického schematu a rozměrových parametrů v závislosti na zkušenostech konstruktéra postupným zkoušením různých konstrukcí a porovnáváním získaných parametrů pomocí poměrně složitých výpočtů.

Automatizace řešení návrhu transformačního bloku je však i zde velice potřebná právě proto, že umožňuje provádět poměrně rychle série potřebných výpočtů a konstruktér může věnovat drahocenný čas větší optimalizaci parametrů transformačního bloku. Tím lze dosáhnout s menšími zkušenostmi v kratším čase optimálnější řešení výsledného transformačního bloku.

Pro komplexní návrh mechanické úchopné hlavice tedy bude třeba spojit předchozí balík výpočtů s novým balíkem týkajícím se výpočtů kinematických parametrů transformačního bloku. Dalším možným nástrojem pro usnadnění návrhu mechanických úchopných hlavic by mohl být grafický editor umožňující kontrolovat soudržnost mechanismu v celém rozsahu zdvihu motoru. Pokud by byl tento systém začleněn do již existujícího CAD systému s možností návaznosti na další podobné nástroje, mohlo by dojít k podstatnému zvýšení produktivity práce konstruktéra odbouráním neproduktivní netvůrčí práce potřebné pro provádění mnoha nezbytných rutinních výpočtů. Nepřímo tak dochází i ke zvýšení užitných hodnot, kvalitativních

parametrů, snížení hmotnosti, energetické náročnosti a mobilnosti výsledné úchopné hlavice.

3.2 Rozbor prostředků automatizovaného návrhu

V současné době absolvují prostředky automatizovaného návrhu revoluční vývoj způsobený prudkým vývojem výpočetní techniky a to nejen v oblasti možností, ale i hromadného nasazení. V této oblasti se v minulosti využívalo pouze velmi složitých a drahých sálových počítačů vybavených programovým vybavením náročným na znalosti a zkušenosti obsluhy. Tyto prostředky automatizovaného návrhu si mohly dovolit pouze velké firmy především z oblasti automobilního a leteckého průmyslu. Vývojem programového vybavení se zabývaly na jedné straně specializované týmy programátorů u firem vyrábějících počítače, které produkovaly především základní grafický software. Na druhé straně se vývojem zabývaly týmy specialistů přímo u jednotlivých uživatelů, které produkovaly naopak na základě dodaných grafických systémů aplikační software specializovaný výhradně na danou oblast činnosti. Takto probíhal vývoj například u firem DEC a IBM na straně základního grafického software a Boeing či Spolku výrobců automobilů v SRN na straně speciálních aplikací.

Prudký rozvoj výpočetní techniky na konci sedmdesátých let, ale především pak v osmdesátých letech, způsobil nový směr vývoje automatizovaného návrhu. Počítač se stal cenově přístupný na straně jedné pro větší počet uživatelů, kteří ho používali především pro hromadné zpracování dat a na straně druhé pro nově vznikající firmy specializující se pouze na vývoj programového vybavení. Ceny počítačů klesaly takřka úměrně se vzrůstem jejich spolehlivosti. Nastal zprvu pozvolný vývoj programových nadstaveb více či méně obecných a teprve jejich praktické využití ukázalo směr dalšího vývoje k dnešnímu stavu.

Nyní se nalézáme v situaci, kdy již ve světě existuje mnoho různých návrhových systémů na různém stupni dokonalosti pro velmi rozličné uplatnění. Dále se budeme zabývat pouze některými obecnějšími návrhovými systémy pro kategorii osobních počítačů, které jsou však zpravidla použitelné (jejich mutace) i pro kate-

gorii pracovních stanic.

Pokud bychom se chtěli zorientovat v současné situaci na trhu CAD systémů pro osobní počítače, zaujmou nás především čtyři ve světě nejrozšířenější systémy. Podívejme se tedy stručně na vzájemné přednosti a nevýhody vývojových verzí, které jsou v současné době nabízeny.

AutoCAD Verze 11:

Výrobce: Autodesk Inc., 2320 Marinship Way, Sausalito, CA 9495

Výhodou tohoto produktu jsou především výkonné prvky, nové možnosti objemového modelování, poměrně vysoká rychlost a otevřená architektura. Nevýhodou je horší úroveň dokumentace. AutoCAD není vázán příliš na žádné konkrétní prostředí ani z hlediska hardware, ani z hlediska oblasti využití. Verze 11 zlepšuje uživatelské rozhraní, výrazně se rozšířily trojrozměrné prvky, standardní součásti programového bloku jsou již i možnosti objemového programování. Tuto verzi zdobí i nové programové rozhraní ADS (AutoCAD Development System - Vývojové prostředí AutoCADu), které v brzké době způsobí rychlý rozvoj dalších aplikací vyvinutých pro AutoCAD v jazyce C či C++ (objektová varianta jazyka C), neboť v tomto jazyce si lze dovolit i náročnější operace, než umožňuje i nadále AutoCADem podporovaný programovací jazyk AutoLISP.

Microstation PC Verze 4.0:

Výrobce: Intergraph Corp., Huntsville, AL 35894-0001

Microstation PC pracuje a celkem slušnou rychlostí a především umožňuje spojení s databázemi Oracle a dBase, což dává programu téměř neomezené databázové možnosti. Nevýhodou je slabší podpora tiskáren a neexistující překlad formátu IGES. Microstation PC je zvláště pružný ze dvou hlavních důvodů - pro své rozhraní a bohatství nástrojů, jež jsou pro něj k dispozici. Nové uživatelské rozhraní je vyjimečný ikonový systém známý také z CADů provozovaných na pracovních stanicích, který činí používání Microstationu PC mimořádně snadným pro začátečníky a přesto nepostihuje zkušené uživatele.

CADKey 386 Verze 4:**Výrobce:** CADKey Inc., 440 Oakland St., Manchester, CT 06040

CADKey je vhodný pro návrhy v mechanice a má slušné prostředí i pro trojrozměrné kreslení, ale současně nepohodlné zobrazování a slabé možnosti atributů. CADKey však postrádá některé zvláštní prvky, jimiž disponují AutoCAD a Microstation PC. Rozhraní CADKey dosahuje pružnosti zkratkovými klávesami a přehlednými menu. Běžný uživatel by měl být dříve schopen používat některé programovací možnosti v CADKey na rozdíl od jiných zde uvedených produktů.

VersaCAD/386 Verze 6.0a:**Výrobce:** Computervision, 100 Crosby Drive, Bedford, MA 01730

K přednostem tohoto CADu patří převodníky pro IGES a DXF formáty, je však poněkud pomalý a vůbec nepodporuje síťové nasazení. VersaCAD příliš nepokračuje ve vývoji, obsahuje však většinu možností očekávaných od předního CADu, chybí mu však podpora sítí a vhodný prostředek pro asociativní kótování. Dvojrozměrné kreslení a trojrozměrné modelování jsou odděleny jako samostatné moduly, přenos dat mezi nimi je však vcelku snadný.

S těmito uvedenými CADy byly podle [8] prováděny důkladné testy rychlosti a spolehlivosti a posuzovány jejich možnosti v těchto oblastech: Kreslicí, editační a umisťovací nástroje, programovatelnost, možnosti vstupu a výstupu, trojrozměrné prvky, práce v síti, možnosti atributů, úroveň dokumentace, obtížnost zvládnutí, snadnost používání, podpora (technická, poradenská). Výsledky provedených testů ukazuje následující tabulka. Znamka se uděluje v rozmezí 0 - 10 (0 - zcela nevyhovující, 10 - ideální produkt).

Produkt:	Verze:	Znamka:
AutoCAD	Verze 11	6.5
Microstation PC	PC Verze 4.0	6.4
VersaCAD/386	Verze 6.0a	6.1
CADKey 386	Verze 4	4.9

Další oblastí, která s návrhovými systémy úzce souvisí je problematika jejich

vzájemné komunikace, a to jak mezi CADy stejné kategorie, tak i mezi velice rozdílnými prostředky, neboť pokud probíhá v současné době vývoj jednotlivých systémů, vzniká potřeba jejich vzájemné interakce a to nejen z důvodů transformace již existující databáze dokumentace na nový systém, ale především z důvodů interakce různých grafických systémů v rámci rozsáhlých podnikových systémů.

Vývoj grafických standardů i standardů pro přenos dat probíhá již více než 10 let. Toto úsilí nabývá na významu právě se snahou o integraci jednotlivých modulů CAX. Důvodem pomalejšího postupu prací je komplexní problematika a široká aplikační oblast. Se zvyšujícím se počtem používaných systémů je to však jediná cesta k udržení přenositelnosti dat i programových produktů.

Většina aplikací počítačové grafiky je velmi náročná na impementaci, která je úrovní náročnosti srovnatelná s impementací programovacího jazyka. V průběhu vývoje jak technických, tak programových prostředků vznikla celá řada grafických systémů, které byly orientovány na určité typy, nebo přímo na jednotlivé druhy počítačů. Od poloviny 70. let se datují první snahy o definici abstraktního grafického pracoviště, které řeší problém přenositelnosti programového vybavení tím, že konkrétní fyzické zařízení je nahrazeno zařízením logickým. Logické V/V zařízení převádí požadavky logických V/V operací vysílaných grafickým systémem na příkazy pro ovladače fyzických zařízení instalovaných v daném operačním systému. V takovém grafickém systému lze nalézt 5 logických vrstev, pro něž jsou definována komunikační rozhraní uvnitř systému.

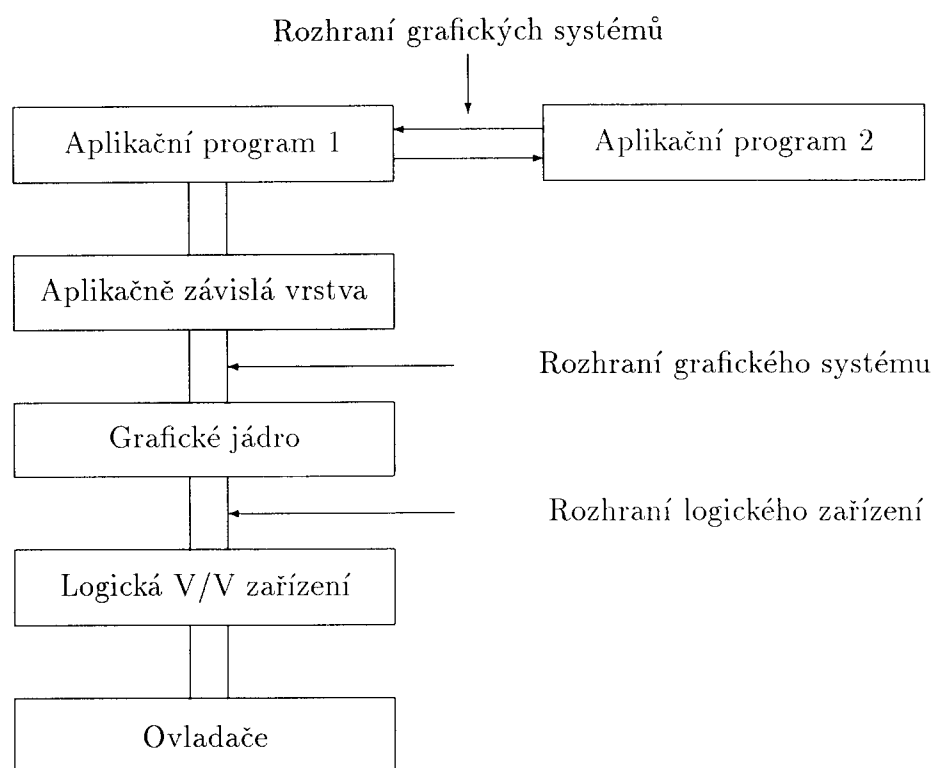
Standardizační úsilí se zde soustřeďuje na definici grafického systémového jádra, které je nezávislé na technických prostředcích. V žádném případě nejde o vytvoření nějakého programového produktu, ale o specifikaci pro samotné impementace různých grafických systémů.

Standardizační úsilí je doprovázeno i řadou problémů:

- snížení výkonu systému, které vyplývá z používání různých funkcí standardizujících formáty přenášených dat (pomalejší zpracování, vyšší paměťová náročnost atd.)

- nedostatečná rychlost reakce v aplikacích, které pracují v reálném čase
- nutnost implementovat celý standard nebo jeho definovanou podmnožinu, čemuž se brání především výrobci

Struktura grafického zařízení:



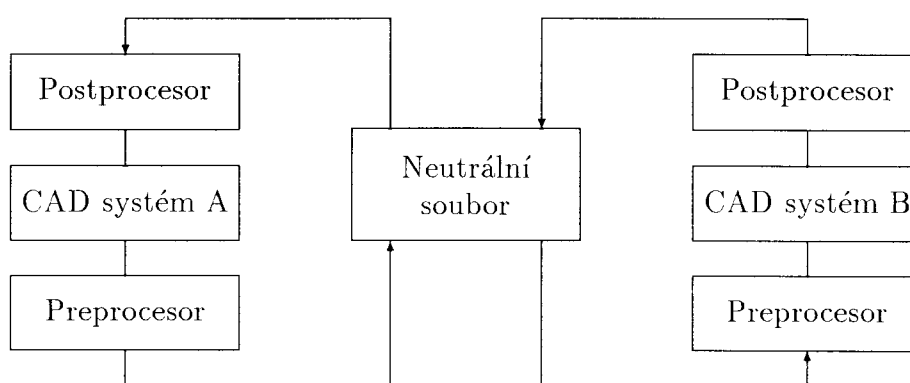
Obr. 4:

Problém výměny dat mezi grafickými systémy (CAD, CADD) je velice široký. Nový přístup lze vysledovat v souvislosti s integrací jednotlivých CAX do systému CIM. Cílem zde není pouze předávání dat, ale spíše předávání (výměna) informací zprostředkovávané právě přenosem dat.

Přenos dat mezi jednotlivými systémy CAD lze v zásadě realizovat třemi způsoby:

1. Klasická dokumentace (výkresy, postupy atd.)
2. Přímý přenos dat (nutnost vytvoření $n \cdot (n-1)$ konverzních modulů nutných pro vzájemnou komunikaci jednotlivých systémů)
3. Použití neutrálního souboru

Způsob výměny dat mezi dvěma systémy CAD při použití neutrálního souboru probíhá takto:



Obr. 5:

Neutrální soubor může mít některý z následujících formátů:

IGES je vyvíjen od r. 1979 firmami Boeing a General Electric, přijat jako ANSI standard v r. 1981, dnes pokračuje další vývoj IGES-4 (1989). Vytváří rozsáhlé datové soubory. Starší koncepce.

SET, VDA-IS tyto standardy jsou založeny na modelu IGES. SET byl přijat v roce 1983 jako národní norma ve Francii. VDA-IS vznikl z iniciativy spolku výrobců automobilů v SRN (1987)

VDA-FS v r. 1986 přijat jako norma DIN a je využíván z důvodu menšího možného počtu entit pouze pro jednodušší procesory především v automobilovém průmyslu SRN.

PDES - v r. 1984 vzniká tento standard , který se snaží vyhnout chybám IGES, přestože čerpá ze stejných základů

PDDI - vznikl na popud americké armády a je navržen tak, že podporuje automatické generování programů pro číslicově řízené stroje.

STEP - mezinárodní normalizační organizace ISO začala pracovat v roce 1984 na standardu, který by umožnil přenos komplexní informace o výrobku. Přestože práce na vývoji tohoto standardu nejsou zatím ani z daleka dokončeny, již nyní je jasná strategie budování a vývoje standardu grafického rozhraní, to se netýká pouze dat popisujících výrobek ve fázi návrhu nebo výroby, ale i dat týkajících se „života“ výrobku (geometrie výrobku, tolerance, materiály, technologie montáže, funkční souvislosti částí výrobku atd.) Popis výrobků ve standardu STEP je rozdělen do 3 vrstev (aplikační, logická, fyzická).

DXF, ISIF - Tyto dva formáty sloužící k přenosu grafické informace nejsou součástí normalizačních snah a vyplývají pouze z pozice výrobců CAD, v tomto případě Autodesku a Intergraphu, na světovém trhu.

Z právě uvedeného plyne, že pokud vybíráme systém CAD, měli bychom mít na zřeteli především jeho kompatibilitu. Tato kompatibilita se projevuje v následujících oblastech:

- možnost zpracování již pořízené dokumentace
- spolupráce s dalšími subsystemy v rámci systému CIM
- otevřený systém schopný dalšího rozvoje
- nároky na obsluhu a potřebnou kvalifikaci pracovníků

4 Program pro automatizovaný návrh MÚH

4.1 Výběr prostředku pro automatizovaný návrh

Podle provedeného podrobného rozboru prostředků automatizovaného návrhu byl pro aplikaci návrhového systému mechanických úchopných hlavíc vybrán návrhový systém AutoCAD, verze 11. AutoCAD verze 11 obsahuje dva následující možné způsoby tvorby aplikací: AutoLISP a ADS.

AutoLISP je implementací programovacího jazyka LISP. Jedná se o vyšší programovací jazyk uzpůsobený především pro grafické aplikace. Umožňuje uživatelům AutoCADu a programátorům tvořit makra a funkce v jazyce vyšší úrovně. LISP je výhodný především pro snadnou implementovatelnost. Tento jazyk byl vybrán i pro výzkum a vývoj umělé inteligence a expertních systémů. Interpret LISPu je vhodný pro nestrukturovanou interakci, která charakterizuje právě proces počítačového navrhování.

Vývojový systém AutoCADu (AutoCAD Development System) je prostředí pro tvorbu aplikací pro AutoCAD v jazyce C. Aplikace ADS není vytvářena jako samostatně stojící program, ale jako soubor externích funkcí, které načítá a spouští interpret AutoLISPu. Aplikace ADS bývají většinou výkonnější co se týče rychlosti a potřeby paměti. Mohou rovněž přímo obsluhovat některé AutoLISPem nedosažitelné prostředky, jako například operační systém a hardware. Tím jsou dobře uzpůsobeny pro aplikace, které vyžadují množství výpočtů a interakce s operačním prostředím.

AutoLISP lépe vyhovuje aplikacím, kde má cena vývoje a úprav větší váhu než efektivnost. To je často případ vysoce interaktivních aplikací, jejichž rychlost stejně omezuje čekání na uživatelské vstupy nebo na provedení operace AutoCADem. Realizace návrhového systému v ADS by byla sice poněkud rychlejší a elegantnější, ale bohužel spustitelná pouze v prostředí AutoCADu verze 11 a vyšší. Právě pro širší použitelnost i v nižších verzích AutoCADu, které obsahují pouze interpret AutoLISPu, byl celý produkt vytvořen v jazyce AutoLISP.

4.2 Požadavky na program

Účelem této práce je podle podkladů z diplomové práce Františka Fialy vytvořit programový produkt, který by měl urychlit a usnadnit komplexní návrh typů mechanických úchopných hlavice určených pro rotační součásti.

Představa o práci tohoto programu je následující:

1. Uživatel programu nadefinuje předmět manipulace, program vypočítá hmotnost, moment setrvačnosti a navrhne optimální místo pro uchopení.
2. Uživatel si vybere kinematické schema a úchopné prvky podle nabídky a bude modifikovat rozměry jednotlivých prvků kinematického schematu.
3. Uživatel provede předběžné zadání motoru.
4. Program provede animaci procesu uchopení, při které budou probíhat výpočty jednotlivých parametrů uchopení.
5. Program provede kontrolu motoru podle zadané úchopné síly pro zadaný rozsah uchopovaných průměrů.
6. Uživatel může provést nový návrh motoru (pokud nevyhověl předběžný návrh) na základě hodnot vypočtených při minulé animaci.
7. Uživatel navrhne průměry čepů a program provede jejich kontrolu opět podle hodnot z předchozí animace.
8. Pokud všechny části vyhovují, může přikročit uživatel k návrhu skutečné hlavice. Pomocí AutoCADu dokreslí ke schematu hlavice buď pouze základní obrysy, nebo celou hlavici.
9. Program provede animaci s reálnými součástkami hlavice, přičemž se projeví případné kolizní stavy skutečné hlavice. Uživatel může dále upravovat jednotlivé součásti tak, aby ke kolizím nedocházelo.
10. Pokud je již hlavice hotova, program vykreslí vybrané závislosti do grafů.

Z kterékoliv části návrhu bude možné vrátit se k předchozím krokům návrhu a modifikovat některé části, pokud se ukáže nevhodnost předchozího návrhu.

Celý programový produkt bude začleněn do systému AutoCAD a jeho produktem budou podklady pro zhotovení sestavných výkresů.

Další použití programu je možné pro úpravy a kontrolu úchopných hlavíc již dříve nakreslených.

Program bude realizován pro nejčastěji používané kinematické schema mechanické úchopné hlavice s pákovými převody, otočně uloženými výstupními čelistmi buď s prizmatickými nebo kruhovými úchopnými prvky.

4.3 Popis programu

4.3.1 Rámcový popis programu

Program návrhového systému mechanických úchopných hlavíc je vytvořen v programovacím jazyce AutoLISP a je určen výhradně pro práci v prostředí grafického editoru AutoCAD. Tento systém je vytvořen tak, aby byl schopen pracovat s minimálními nároky na paměť na počítačích s velikostí operační paměti 1MB, tedy aby byl schopen pracovat i bez použití externího procesoru AutoLISPu.

Z hlediska uživatele je celý systém začleněn do struktury menu AutoCADu, takže umožňuje nejen sekvenční styl návrhu úchopných hlavíc, ale především dodává AutoCADu mnoho vzájemně propojených nástrojů umožňujících komplexní návrh mechanických úchopných hlavíc a to nejen z kinematického, ale především z pevnostního hlediska. Stejně tak prostřednictvím své variability umožňuje nejen kontrolu, ale i modifikace již vytvořených skutečných úchopných hlavíc.

Tento systém lze využít i pro studium struktur kinematických řetězců mechanických úchopných hlavíc, a to především díky možnostem generování grafů závislostí jednotlivých charakteristických parametrů v souvislosti se změnami umístění jednotlivých uzlových bodů, poměrů délek, popřípadě úhlů.

Všechny výpočtové vztahy použité v programu vycházejí ze studie uvedené v diplomové práci Františka Fialy, kde jsou také velmi podrobně popsány a zdůvodněny, proto se budeme nadále věnovat pouze programu samotnému.

Celý návrhový systém obsahuje v současné době vztahy pouze pro kinematické schema hlavice typu A podle [4] s prizmatickými a kruhovými úchopnými prvky.

Celý systém je budován tak, aby umožňoval další rozvoj, popřípadě interakci s dalšími podobnými návrhovými systémy v prostředí AutoCADu. V každém okamžiku jsou v paměti přítomny pouze ty funkce a procedury, které jsou pro danou činnost nutné, čímž je vlastně umožněna činnost i ve stísněných paměťových podmínkách. Celý program a koordinace jednotlivých činností jsou řízeny prostřednictvím uživatelského menu HLAVICE. Jednotlivé procedury mezi sebou komunikují prostřednictvím globálních proměnných, které lze kdykoliv společně s aktuálním výkresem uložit a použít někdy v budoucnosti. Pro urychlení práce, hlavně při inicializaci je připraven speciální prototypový výkres ACAD.DWG, který obsahuje potřebné definice používaných prvků včetně připravených formulářů pro jednotlivé grafy. Všechny nově definované procedury a funkce stejně jako i globální proměnné jsou popsány v další kapitole.

Návrhový systém Hlavice se inicializuje při první volbě položky HLAVICE v základním menu AutoCADu. Nejprve se provede kontrola zda výkres obsahuje potřebné prvky a pokud ne, tak si je program před další činností vygeneruje. Následuje načtení konfiguračního souboru a pokud existuje datový soubor shodného jména se jménem aktuálního výkresu, provede se i načtení hodnot globálních proměnných z tohoto souboru.

Od této chvíle lze používat všechny nástroje popsané v dalších kapitolách, přičemž v každém menu či submenu je přístupná položka POMOC, po jejímž zvolení se do aktuálního okna grafického editoru zobrazí snímek s nápovědou k jednotlivým položkám konkrétního menu.

Jednotlivé části kresby (schema, motor, úchopný průměr, grafy atd.) jsou umístěny ve zvláštních vrstvách a lze je kdykoliv buď zobrazit, nebo vypnout. Ani-

mace se provádějí pomocí modifikace parametrů jednotlivých entit prvků schematu či hlavice a tím pádem probíhají i v případě, že je právě daná vrstva vypnuta. Jednotlivé prvky kresby (čáry, oblouky atd.) jednotlivých součástí skutečné hlavice jsou spojeny do bloků a jejich animace probíhá změnou parametrů entit těchto bloků. Před každou animací je provedeno načtení aktuálních hodnot koncových bodů entit přímek (oblouku) schematu, čímž se průběžně aktualizuje nastavení proměnných souřadnic uzlových bodů kinematického schematu. Pozor, osa symetrie hlavice musí souhlasit s x-ovou osou globálního souřadného systému.

Podrobné pokyny pro instalaci a všechny potřebné soubory jsou včetně komentovaných zdrojových souborů a posledních změn umístěny na disketě, která je součástí přílohy této diplomové práce

4.3.2 Stručný popis globálních proměnných

- * proměnná se ukládá do datového souboru
- * proměnná se ukládá do konfiguračního souboru

ALFA1-6	natočení základní čáry horní části schematu vůči ose x
BARVAR	* číslo barvy aktuální varianty
BETA	úhel mezi prvky 4 a 5
BETA2	úhel mezi prvky 5 a 6
BETA3	úhel rozevření prizmatických čelistí
CEMKM	* celková hmotnost hřídele
CESTADWG	* cesta do adresáře *.dwg (c:\acad\dwg\)
CESTAHLV	* cesta do adresáře *.hlv (c:\acad\hlv\)
D	uchopovaný průměr
DMAX	* maximální uchopovaný průměr
DMIN	* minimální uchopovaný průměr
DOPT	* optimální (požadovaný) uchopovaný průměr
DPI	* průměr pístu
DROZ	délka rozevření prizmatických čelistí

DX1	delta x11 - grafická
DX2	delta x11 - třením
EN1 ... EN5	entity horní poloviny schematu
EN1S ... EN5S	entity spodní poloviny schematu
ENB1 ... ENB5S	entity bloků těles
ENK	entita kružnice uchopovaného objektu
ENP	první/poslední pracovní entita (nacti/naccel)
ENPIST	entita pístu
FC3	* maximální síla v čepu 3
FC4	* maximální síla v čepu 4
FC5	* maximální síla v čepu 5
FI	kontaktní úhel
FM	síla na motoru (vypočtená)
FMZ	* síla na motoru (zadaná)
FTR	* součinitel tření
FU	úchopná síla (vypočtená)
FUZ	* úchopná síla (zadaná)
GRAFY	seznam názvů grafů
GRMAXX	maximální hodnoty grafů na ose x
GRMAXY	maximální hodnoty grafů na ose y
GRMINX	minimální hodnoty grafů na ose x
GRMINY	minimální hodnoty grafů na ose y
GRVRSTVA	aktuální grafická vrstva
GXD	* x-ové souřadnice počátku výkresových ploch
GYD	* y-ové souřadnice počátku výkresových ploch
HLAVICE	T ... byl proveden hlsetup
INTERP	* "d" "s" "f" - typ interpolace
JEGRAF	* (0 1 0 1 1 0 1 1 1 1) které grafy kreslit
KROKU	* počet kroků při kreslení grafů
L1...L7	délky čar
MEMOTOR	T jsem v menu motor (nil nejsem)

MI	transformační úhel
MINBOD	seznam posledních bodů křivek grafů
MINENT	seznam entit křivek grafů
NAZVAR	★ název varianty
PC3	★ průměr čepu 3
PC4	★ průměr čepu 4
PC5	★ průměr čepu 5
PDC	★ dovolený tlak v čepch (otlačení)
R	poloměr kruhové části čelisti
R0	uchopovaný poloměr
SC3	★ šířka čepu 3
SC4	★ šířka čepu 4
SC5	★ šířka čepu 5
SULOZDAT	seznam ukládaných dat (*.hlv)
TLGRAf	★ tloušťka grafů
TLP	★ tlak v rozvodu pro motor
TYPCEL	★ vybraný typ úchopných prvků
TYPMOT	★ 1-2 typ motoru
UHCEL	úhel kruhové části čelistí
VARIANTA	★ číslo varianty
VYBR	★ 1-3 číslo vybraného schematu
VZTAHX	vztahy pro výpočet x-ových souřadnic grafů
VZTAHY	vztahy pro výpočet y-ových souřadnic grafů
X	zdvih motoru ($X1 = X0D + X$)
X0D	★ x-ová souřadnice dolní úvrati
X0H	★ x-ová souřadnice horní úvrati
X1...X11	x-ové souřadnice bodů
XU	★ vzdálenost bodu uchopení odleva
Y1...Y11	y-ové souřadnice bodů
Z1,Z2,Z3	znaménka
ZUB1-5	★ úhel natočení přímky při definici bloku tělesa

4.3.3 Stručný popis nově definovaných funkcí

ACOS	arccos
ANIMCEL	animace (vlečení) hlavice podle čelistí (bodu 6)
ANIMMOT	animace (vlečení) hlavice podle zdvihu motoru
ANONE	dotaz ano/ne s přednastavením implicitní hodnoty
ASIN	arcsin
CELNAPR	sevření čelistí na daný průměr
CEPY	nastavení průměru, šířky čepu a dovol. tlaku
COTG	cotangens
DEFPOHL	definice pohledů pro výrobu prototypového výkresu
DEJHOD	vstup float hodnoty s přednastavenou implicitní hodnotou
GRAF	kreslení grafů (posledních úseků od minulých koncových bodů jednotlivých křivek)
GRAFIN	inicializace grafů před zahájením jejich vykreslování
GRAFMOT	kontrola, animace a vykreslení grafů podle posunu motoru
HLHELP	zobrazí snímek daného čísla, vyčká na stisk klávesy <ENTER> a vrátí původní obraz
HRIDEL	zadání uchopovaného objektu - hřídele, výpočet uchopné síly, těžišť a dalších setrvačných parametrů
KRESAN	vykreslení skutečného schématu podle právě vypočtených souřadnic, aktualizace kružnice uchopeného průměru a změna polohy skutečných těles součástí hlavice
KRESANCEL	vykreslování čelistí podle právě vypočtených souřadnic
KRESCEL	prvotní vykreslení vybraných čelistí do schématu
KRESHLAS	provede inicializaci výpisu hlášení a nastaví příslušné proměnné entit
KRESHRID	prvotní vykreslení hřídele
KRESMRIZ	kreslení mřížovaného podkladu grafů
KRESOSY	kreslení os

KRESPUV	vykreslení původního schematu
MAXMOT	nastavení motoru na horní úvrať
MAXPRUM	nastavení čelistí na maximální uchopovaný průměr
MINMOT	nastavení motoru na dolní úvrať
MINPRUM	nastavení čelistí na minimální uchopovaný průměr
NACCEL	načtení aktuálních parametrů úchopných prvků
NACTDATA	načtení výkresu a uložených proměnných ze souboru
NACTI	nastavení aktuálních parametrů schematu podle současného stavu výkresu
NACTICFG	načtení konfigurace z konfiguračního souboru
NASTAV	vrátí upravený seznam o nastavenou hodnotu (opak nth)
NEGRAFY	smaže všechny grafy
NELAYER	zrušit všechny prvky zadané vrstvy
NEZ18	zrušit všechny procedury ze Z18.LSP
NEZ2	zrušit všechny procedury ze Z2.LSP
NEZ5	zrušit všechny procedury ze Z5.LSP, z11 (popř.Z12)
NEZ6	zrušit všechny procedury ze Z6.LSP
NEZ7	zrušit všechny procedury ze Z7.LSP
OPTIMAL	nastavit hlavici na optimální uchopovaný průměr
POSUNBOD	modifikace poloh základních bodů schematu
POSUNMOT	nastavit hlavici na zadaný posun motoru
PRNACCEL	první načtení entit úchopných prvků
PROTOTYP	výroba prototypového výkresu
ROZBITEL	rozbití vybraného bloku tělesa
ROZBIVSE	rozbití všech bloků již definovaných těles
TAN	tangens
TELESA	vytvoření bloku dílu skutečné hlavice
TISKGR	tisknout / kreslit vybraný graf
ULOZCFG	uložení konfigurace
ULOZDATA	uložení výkresu a obsahu proměnných
UPKR	upravit křivku - změna interpolace

UPRCEL	modifikace čelistí
VARI	zobrazí požadovanou variantu grafu
VYBERGR	výběr grafu pro následné kreslení
VYPCEL	výpočet souřadnic pro animaci podle čelistí (B6)
VYPCELM	výpočet souřadnic čelistí podle rozevření
VYPCELPR	výpočet souřadnic čelistí podle průměrů
VYPMOT	výpočet souřadnic schematu v závislosti na posunu motoru
VYPOCGR	výpočet hodnot pro grafy
ZOBRGRAF	zobrazit graf podle zadaného čísla

4.3.4 Stručný popis obsahu zdrojových souborů

Zdrojové soubory pro AutoLISP:

NÁZEV DEFINOVANÉ PROCEDURY

⇒ činnost

Z1.LSP HRIDEL, KRESHRID

⇒ zadání a výpočty týkající se uchopovaného objektu, spuštění a zrušení

Z2.LSP VYPMOT, VYPCEL, KRESAN, NACTI, PRNACCEL, NEZ2

⇒ veškeré výpočty kinematických veličin, procedury pro načtení aktuálního základního schematu a jeho zobrazení podle vypočtených souřadnic přítomné neustále v paměti, načtení Z8.LSP, popř. Z9.LSP v závislosti na typu úchopných prvků

Z3.LSP POSUNBOD, UPRCEL

⇒ procedury pro modifikaci schematu

Z5.LSP KRESPUV, NEZ5

⇒ prvotní vykreslení schematu, načtení Z11.LSP, popř. Z12.LSP v závislosti na typu úchopných prvků, spuštění a smazání

Z6.LSP ANIMMOT, ANIMCEL, CELNAPR, POSUNMOT, MINMOT, MAXMOT, MINPRUM, MAXPRUM, OPTIMAL, NEZ6

⇒ procedury pro menu ANIMACE a MOTOR

Z7.LSP TELESA, ROZBITEL, ROZBIVSE, NEZ7

⇒ procedury pro menu TĚLESA

Z8.LSP VYPCELM, VYPCELPR, NACCEL, KRESANCEL

⇒ procedury pro výpočty, načtení a vykreslení úchopných prvků podle vypočtených hodnot pro kruhové úchopné prvky

Z9.LSP VYPCELM, VYPCELPR, NACCEL, KRESANCEL

⇒ procedury pro výpočty, načtení a vykreslení úchopných prvků podle vypočtených hodnot pro prizmatické úchopné prvky

Z11.LSP KRESCEL

⇒ prvotní vykreslení kruhových úchopných prvků do schematu

Z12.LSP KRESCEL

⇒ prvotní vykreslení prizmatických úchopných prvků do schematu

Z14.LSP žádná

⇒ výpočet motoru podle síly, volá Z16.LSP

Z15.LSP žádná

⇒ výpočet motoru podle průměru pístu, volá Z16.LSP

Z16.LSP žádná

⇒ vykreslení schematu motoru po Z14.LSP či Z15.LSP

Z17.LSP VYBERGR

⇒ výběr grafu pro kreslení, spuštění a smazání

Z18.LSP ULOZDATA, NACTDATA, ZOBRGRAF, TISKGR, NEZ18

⇒ procedury pro menu POMŮCKY

Z19.LSP PROTOTYP, NACTICFG, ASIN, ACOS, TAN, COTG, DEJHOD, ANONE,
NELAYER, UPKR, VARI, NASTAV, VYPOCTY, NEZSCH, HLHELP

⇒ výroba prototypového výkresu, nastavení důležitých proměnných, inicializace vývojového systému HLAVICE a definice nejčastěji používaných procedur

Z20.LSP KRESMRIZ

⇒ vykreslení podkladových sítí grafů, spuštění a smazání

Z21.LSP GRAFMOT, VYPOCGR

⇒ procedury pro kreslení grafů, volá Z29.LSP a Z30.LSP, spuštění a smazání

Z22.LSP ULOZCFG

⇒ uložení konfigurace, spuštění a smazání

Z23.LSP KRESOSY

⇒ vykreslení os grafů, spuštění a smazání

Z24.LSP NEGRAFY

⇒ smazání všech variant grafů, spuštění a smazání

Z25.LSP KRESHLAS

⇒ inicializace textu výpisu důležitých hodnot, spuštění a smazání

Z26.LSP DEFPOHL

⇒ inicializace pohledů, spuštění a smazání

Z27.LSP CEPY

⇒ zadání parametrů čepů, spuštění a smazání

Z28.LSP TESTCEP

⇒ kontrola čepů, spuštění a smazání

Z29.LSP GRAFIN

⇒ pomocná inicializační procedura pro Z21.LSP, spuštění a smazání

Z30.LSP GRAF

⇒ pomocná procedura pro kreslení grafů

Ostatní důležité soubory:

ACAD.MNU Základní menu AutoCADu s odkazem na menu HLAVICE

HLAVICE.MNU Menu návrhového systému HLAVICE

START.SCR Startovací sekvence

KONEC.SCR Ukončovací sekvence

HLAVPRAC.SCR Pracovní script pro tisk,plot,export

xxxxxxxx.DWG Výkres

xxxxxxxx.HLV Důležitá data k výkresu

HLAVICE.DWG Poslední obraz před odchodem

HLAVICE.HLV Poslední data před odchodem

HLAVICE.CFG Konfigurace návrhového systému HLAVICE

HLAVICE.BAT Dávkový soubor pro spuštění AutoCADu s nastavením pro návrhový systém HLAVICE

HLAVICE.SLB Knihovna snímků pro návrhový systém

4.3.5 Seznam dalších definovaných prvků

Vrstvy

HLSHEMA Kinematické schema úchopné hlavice

HLSCHPOM Pomocné prvky schematu (motor)

HLGR0 Křivky a popisy jednotlivých grafů varianty 0

HLGR2 Křivky a popisy jednotlivých grafů varianty 2

... ..

HLGR9 Křivky a popisy jednotlivých grafů varianty 9

HLUCHPRU Kružnice uchopovaného průměru

HLHODNOT Vypisované hodnoty při pevnostní kontrole

HLPOPOSY Základ grafů (osy + základní popis)

HLMRIZOV Podkladová síť grafů s desetinným dělením

HLOBRUB Ohraničení kreslicích ploch grafů

Bloky

US1 Nadefinovaná skutečná součást příslušející prvnímu prvku základního kinematického schématu hlavice

... ..

US5 Nadefinovaná skutečná součást příslušející pátému prvku základního kinematického schématu hlavice

US1S Nadefinovaná skutečná součást příslušející prvnímu spodnímu prvku základního kinematického schématu hlavice

... ..

US5S Nadefinovaná skutečná součást příslušející pátému spodnímu prvku základního kinematického schématu hlavice

PIST Píst schématu motoru

Styly

hltxt1 Textový styl pro popisy grafů skutečnými hodnotami (standardně kurzíva)

hltxt2 Textový styl pro ostatní popisy grafů a hlášení hodnot při pevnostní kontrole (standardně complex)

Pohledy

POHL1 Jedno okno, zobrazena hlavička

POHL2 Čtyři okna stejné velikosti

POHL3 Čtyři okna stejné velikosti

POHL4 Jedno okno, zobrazen graf

POHL5 Jedno okno velké s hlavičkou a tři malé grafy

POHL6 Dvě menší okna s grafy a jedno větší s hlavičkou

Snímky

S01.SLD Kinematické schema typu 1

S02.SLD Kinematické schema typu 2

S03.SLD Kinematické schema typu 3

C01.SLD Kruhové úchopné prvky

C02.SLD Prizmatické úchopné prvky

INT1.SLD Interpolace - přímky spojující body

INT2.SLD Interpolace - křivka procházející body

INT3.SLD Interpolace - spline křivka

MRIZEJO.SLD Graf se sítí s desetinným dělením

MRIZENE.SLD Graf pouze s osami

OBRUBA.SLD Graf s ohraničením kreslící plochy grafu

POHL1.SLD Pohled POHL1

POHL2.SLD Pohled POHL2

... ..

POHL6.SLD Pohled POHL6

HELP0.SLD Kinematické schema s popisem

HELP1.SLD Help k menu HLAVICE

HELP2.SLD Help k menu ZADÁNÍ

HELP3.SLD Help k menu SCHEMA

HELP4.SLD Help k menu MOTOR

HELP5.SLD Help k menu TĚLESA

HELP6.SLD Help k menu NÁVRH

HELP7.SLD Help k menu KONTROLA

HELP8.SLD Help k menu ANIMACE

HELP9.SLD Help k menu POMŮCKY

HELP10.SLD Help k menu ZOBRAZ

HELP12.SLD Help k menu NASTAV

HELP13.SLD Help k menu GRAFY

CEPY.SLD Označení čepů

GRAF1.SLD Obraz grafu typu 1

... ..

GRAF10.SLD Obraz grafu typu 10

HLAVICE.SLB Knihovna výše uvedených snímků

SNIMKY.SEZ Seznam snímků k vytvoření knihovny HLAVICE

SNIMKY.BAT Dávkový soubor pro vytvoření knihovny snímků

4.3.6 Další rozšíření programu

Jak z již uvedeného vyplývá, jedná se o návrhový systém, který umožňuje provádět kompletní návrh mechanických úchopných hlavíc. Co se týče koncepce tvorby programu, je zpočátku pojata tak, aby bylo možné dále tento systém rozvíjet bez potřebných zásahů do již vytvořených programů. Rozšiřování programu je možné na dvou zcela odlišných úrovních.

První možností dalšího rozšíření je možnost vytvoření dalších nástrojů jako například výběr typizovaných motorů, úchopných prvků a dalších stavebních celků z databáze již vyráběných součástí, či poloautomatická optimalizace některých částí kinematického schématu.

Druhá cesta dalšího rozšíření tohoto návrhového systému spočívá v rozvíjení použitelných typů kinematických struktur úchopných hlavíc a úchopných prvků.

V prvním případě je možný další rozvoj bez větších komplikací tím způsobem, že budou další samostatné funkce začleněny do uživatelského menu HLAVICE a odtud volány s ohledem na paměťové nároky již načtených funkcí. Po použití musí být paměť (prostor uzlů) opět od těchto nových funkcí uvolněna. V rámci těchto nových funkcí lze využívat uvedené globální proměnné a uživatelsky definované funkce přítomné v paměti (především funkce obsažené ve zdrojových souborech Z19.LSP, Z2.LSP, Z8(9).LSP), nebo potřebné uživatelsky definované funkce načíst ze zdrojových souborů podle uvedeného seznamu. Opět je třeba dávat velký pozor

na to, aby nebyly v rámci činnosti nově definované funkce uvnitř některého submenu zrušeny funkce, které jsou již načteny pro využití v daném submenu, nebo naopak aby nezůstaly v paměti funkce námi načtené po skončení nově definované funkce.

Podrobné informace o tom, které uživatelsky definované funkce jsou ve kterémkoliv konkrétním okamžiku přítomny v paměti lze snadno vyčíst ze zdrojového souboru HLAVICE.MNU.

V případě rozšiřování návrhového systému o další kinematické struktury mechanických úchopných hlavice je situace také celkem snadná. Pro každý jednotlivý typ struktury je třeba vytvořit zdrojový soubor, který je v podstatě lehce modifikovanou kopií souboru Z2.LSP (výpočtové vztahy, načítání a modifikace základního schematu) při striktním dodržení všech názvů procedur a funkcí, jejich činností a globálních proměnných. Stejným způsobem je třeba vytvořit nové zdrojové soubory, variace na Z8.LSP a Z9.LSP (výpočtové vztahy, načítání a modifikace schemat úchopných prvků pro dané základní schema), dopsat potřebné rozměry nových struktur a úchopných prvků do zdrojových souborů Z5.LSP, Z11.LSP a Z12.LSP (procedury pro vykreslení a prvotní návrh schematu), přidat potřebné ikony do výběrových menu a podle výběru načítat příslušné zdrojové soubory. Při přidávání dalších typů úchopných prvků musíme mít na zřeteli, že výpočtové vztahy pro ně určené se budou se změnou struktury základního schematu lišit a tudíž budou v různých zdrojových souborech pro jednotlivé kombinace struktura-úchopný prvek.

Tím, že zůstávají zachovány názvy procedur a výsledky jejich chování, není potřeba nikterak měnit další zdrojové soubory či procedury celého návrhového systému, čímž je dána pružnost tohoto celého systému i co se týče možností dalšího rozšiřování.

4.4 Návod k použití programu

4.4.1 Postup návrhu mechanické úchopné hlavice

Spustíme grafický editor AutoCAD a vybereme volbu kreslení nového výkresu (volba 1). Je založen nový výkres pod námi zadaným jménem, který již obsahuje některé

předdefinované prvky a entity (pohledy, styly, hladiny, bloky, podklady budoucích grafů atd.).

V obrazkovém menu zvolíme položku HLAVICE. Návrhový systém provede kontrolu, zda doposud neexistuje datový soubor stejného jména, jaké má současný výkres. Pokud takový soubor již existuje, program z něho načte implicitní hodnoty pro další práci, což má význam především pokud musíme práci s návrhovým systémem přerušit, popřípadě miníme dále rozvíjet či modifikovat starší variantu úchopné hlavice.

Zvolíme v submenu HLAVICE položku ZADÁNÍ a provedeme zadání jednotlivých položek.

Pokud známe rozměry uchopovaného hřídelovitého objektu a nemáme ještě vypočtenou potřebnou úchopnou sílu, vybereme položku HŘÍDEL. Podle výzev v příkazové řádce zadáváme postupně jednotlivé hodnoty (délek a průměrů úseků, měrnou hmotnost atd.). Program podle zadávaných hodnot průběžně vytváří schematický náčrt hřídele. Po zadání všech potřebných hodnot vypočte program těžiště hřídele a navrhne ho jako optimální střed uchopení objektu. Pokud nám tento vyhovuje, odpovíme klávesou ENTER, popř. pravým tlačítkem myši. Pokud ne, zadáme novou polohu středu uchopení (platí pouze X-ová souřadnice) a program provede výpočet potřebné úchopné síly.

Jako další krok přistoupíme k zadávání základního kinematického schématu. Zvolíme položku SCHEMA a zobrazí se další submenu.

Nejprve přikročíme k výběru typu modifikace dané kinematické struktury základního schématu spuštěním položky VYBERTYP. Zobrazí se nabídka různých typů (modifikací) kinematických struktur základního kinematického schématu úchopné hlavice, po ní nabídka typů úchopných prvků. Po provedeném výběru zadáme na výzvu optimální (požadovaný), minimální a maximální uchopovaný průměr. Po vykreslení základního schématu nastavíme zbývající rozměrové parametry týkající se daných úchopných prvků.

Pokud byl při startu nalezen datový soubor se stejným názvem jako je ná-

zev současného výkresu, budou všechny hodnoty implicitně nastaveny na předchozí konfiguraci. Pokud se však jedná o první spuštění, dojde k přednastavení těchto hodnot podle rozměrů ostatních již zadaných parametrů.

Pokud máme konkrétní představy o umístěních jednotlivých bodů kinematického schematu, zvolíme položku POSUNBOD, vybereme podle jednoduché nápovědy bod, jehož polohu můžeme měnit a zadáme nové umístění bodu (buď číselně, nebo pomocí grafického kurzoru. Nabídka k posunu bodu se cyklicky opakuje a je ukončena až zadáním čísla bodu 0.

Pokud jsme dosáhli požadované schéma, opustíme toto menu a vrátíme se zpět do menu ZADÁNÍ.

Jako další zadáme důležité parametry motoru. Vybereme tedy položku MOTOR. Zde bude naším cílem nastavit mezní polohy posunutí motoru (dolní a horní úvrať) a provést návrh motoru s výpočtem maximální síly na motoru. Nastavení úvratí provedeme pomocí položek DOL_UVR a HOR_UVR, přičemž jimi označíme aktuální polohu mechanismu jako dolní/horní úvrať motoru. Dolní úvrať se předpokládá vlevo a horní vpravo. Pokud omylem provedeme zadání buď dolní úvratí nad horní a nebo horní úvratí pod dolní, jsou nastaveny do tohoto bodu obě úvratí, toto mějme na paměti, pokud by se nám při návrhu motoru jak je uvedeno dále vykreslovalo schéma motoru pouze jako svislá čára. Pro pohyb se schematem pro dosažení budoucích úvratí použijeme nástroje z aktuálního menu. Tyto nástroje nám umožní nastavit schéma na zadaný průměr, posun členu 1 (posun ještě nevykresleného pístu motoru), ale i animovat schematem pomocí grafického kurzoru, a to opět buď podle posunu členu 1 a nebo rozevření čelistí, přičemž se schéma pohybuje pouze v mezích jeho možností, tak aby nedošlo k roztžení některého členu.

Pokud máme zadánu dolní a horní úvrať, můžeme přikročit k samotnému návrhu motoru.

Zvolíme položku NÁVRH a vybereme si způsob návrhu motoru. Návrh motoru provedeme buď podle požadované síly na motoru nebo podle průměru válce a tlaku tekutiny. V praxi zpravidla navrhujeme průměr válce motoru podle rozměrových parametrů a teprve pokud motor nevyvodí potřebnou sílu pro uchopení (což

poznáme po spuštění pevnostní kontroly, jak je uvedeno dále) přikročíme k návrhu motoru podle potřebné síly. Po zadání patřičných parametrů včetně tlaku tekutiny v rozvodu se vykreslí schema motoru v měřítku 1:1 a vypočte maximální síla na motoru.

Po provedení návrhu motoru se vrátíme zpět do základního menu HLAVICE a vybereme položku KONTROLA.

V první fázi budeme provádět kinematickou kontrolu navrženého mechanismu v rámci submenu ANIMACE pomocí různých nástrojů, které jsou zde obsaženy. Mnoho nástrojů je shodných s již používanými při návrhu motoru. V tomto okamžiku však narozdíl od submenu MOTOR probíhá neustálá kontrola parametrů hlavice včetně motoru, což nám umožní simulovat pohyb skutečného mechanismu. Tato animace je určena především pro odhalení nedostatků týkajících se prostorového uspořádání jednotlivých dílů či délek táhel, tak aby nedocházelo například k „roztrhání“ táhel mechanismu v rámci celého rozsahu zdvihu motoru, popřípadě aby hlavice umožňovala uchopení požadovaných průměrů pokud od okamžiku nastavení úvratí došlo ke změně kinematického schématu.

Pokud mechanismus splňuje nároky na korektní chování v celém rozsahu zdvihu motoru, můžeme přikročit k jeho pevnostní kontrole. V opačném případě se vrátíme do menu ZADÁNÍ a provedeme potřebné korekce na kinematickém schématu mechanismu a kinematickou kontrolu opakujeme.

Před prvním spuštěním pevnostní kontroly provedeme nastavení parametrů vykreslovaných grafů v submenu GRAFY. Nejprve vybereme které grafy budeme chtít generovat (pokud se vykresluje větší množství grafů, práce grafického editoru se výrazně zpomalí). Nabízeny jsou grafy závislosti:

- Úchopná síla vypočtená podle maximální konstantní síly na motoru v závislosti na zdvihu motoru
- Úchopná síla vypočtená podle maximální konstantní síly na motoru v závislosti na uchopovaném průměru

- Potřebná síla na motoru vypočtená podle minimální konstantní úchopné síly v závislosti na zdvihu motoru
- Potřebná síla na motoru vypočtená podle minimální konstantní úchopné síly v závislosti na uchopovaném průměru
- Transformační úhel v závislosti na zdvihu motoru
- Transformační úhel v závislosti na uchopovaném průměru
- Úchylka středu uchopovaného objektu v závislosti na uchopovaném průměru důsledkem grafických závislostí
- Úchylka středu uchopovaného objektu v závislosti na uchopovaném průměru důsledkem důsledkem tření
- Zdvih motoru v závislosti na uchopovaném průměru
- Kontaktní úhel v závislosti na uchopovaném průměru

Dále si zvolíme styl podkladu grafu z následujících variant:

1. pouze osy
2. orámování plochy grafu
3. síť s desetinným dělením pro pozdější snazší odečítání hodnot

vybereme typ interpolace grafů:

1. pouhé spojení vypočítaných bodů přímkami
2. proložení plynulou křivkou procházející vypočítanými body
3. proložení spline křivkou

a nastavíme tloušťku křivek grafů.

Tím máme nastaveny parametry pro vykreslování grafů a můžeme přikročit k samotné kontrole, kterou spustíme výběrem položky KONTROLA v submenu KONTROLA. Nejprve na výzvu provedeme nastavení potřebných parametrů, a to: název varianty, číslo varianty grafů, barva varianty grafů, počet kroků pro výpočet bodů, koeficient tření, maximální konstantní sílu na motoru a minimální konstantní úchopnou sílu, přičemž jsou již přednastaveny implicitní hodnoty buď z minulé kontroly, z datového souboru či navržené systémem jako doporučené. Maximální konstantní síla na motoru je přednastavena na hodnotu vypočtené (popř. zadané) síly při návrhu motoru a minimální konstantní úchopná síla je nastavena na hodnotu vypočtenou při zadávání objektu manipulace.

Poté se spustí automatická animace kinematického schematu hlavice podle posunu motoru z dolní úvrati do horní na zadaný počet kroků, při níž se do části kreslicí plochy k tomu určené vypisují aktuální hodnoty všech důležitých silových parametrů a vykreslují vybrané grafy. Po ukončení animace je provedeno vyhodnocení naměřených hodnot a na textové obrazovce se zobrazí jeho výsledek. Opět jako tomu bylo v případě kontroly kinematických parametrů, pokud hlavice nevyhovuje, vrátíme se na začátek a po úpravě schematu provedeme opět celý postup kinematické i pevnostní kontroly.

Pokud jsme s výsledky měření spokojeni, přikročíme k dalšímu bodu, kterým je kontrola čepů. Vrátime se zpět do submenu ZADÁNÍ a spustíme položku ČEPY. Podle výzev v příkazové řádce nastavíme (popřípadě ponecháme implicitní) patřičné rozměry jednotlivých čepů a povolený tlak. Po zadání těchto parametrů proběhne automaticky kontrola čepů podle extrémních hodnot získaných při minulé pevnostní kontrole mechanismu. Pokud čepy nevyhovují, můžeme změnit jejich rozměry, popřípadě upravit schema a provést znovu již provedený postup kontrol s tím rozdílem, že kontrola čepů bude již probíhat automaticky při vyhodnocování pevnostní kontroly (po vykreslení grafů).

Pozor, pokud nebude pro kreslení vybrán žádný graf, proběhne při pevnostní kontrole pouze animace schematu s vypisováním aktuálních hodnot jednotlivých

parametrů a závěrečné vyhodnocení, čímž se prováděná pevnostní kontrola podstatně zrychlí a potřebné grafy lze vykreslit teprve až bude hlavice pevnostně vyhovující. Pokud se však kreslení grafů uskuteční, budou se grafy vykreslovat do další samostatné hladiny, přičemž získáme možnost následně jednotlivé grafy porovnávat a na základě porovnání dále optimalizovat úchopnou hlavici s cílem upřednostnit některý z charakteristických parametrů hlavice (například minimalizovat odchylku středu uchopovaného objektu v určitém rozsahu uchopovaných průměrů).

Při vykreslování grafů nejsou pro větší přehlednost předchozí grafy zobrazeny, ale lze je prohlížet pomocí roletového menu buď jako jednotlivé grafy, nebo je vzájemně překrývat, současně je registrováno maximálně deset variant grafů (0-9), které lze výhodně barevně odlišit, přičemž v rámci jedné varianty může být umístěno všech výše uvedených deset typů grafů.

Pokud dosavadní produkt splňuje všechny požadavky (absolvoval úspěšně všechny kontroly), může tím naše práce skončit. Získali jsme v poměrně krátkém časovém okamžiku kinematické schéma požadované úchopné hlavice v měřítku 1:1 s provedenou pevnostní kontrolou čepů a navrženými parametry motoru.

Pravděpodobně však budeme chtít zhotovit kompletní dokumentaci pro výrobu dané úchopné hlavice, k čemuž můžeme využít dalších služeb tohoto návrhového systému.

K již vytvořenému schématu úchopné hlavice (včetně motoru a úchopných prvků) můžeme dokreslit pomocí bohatých nástrojů grafického editoru AutoCAD buď pouze obrysy, a nebo již skutečný výkres úchopné hlavice kreslený tak, že je schéma postupně „obaleno“ skutečnými součástmi.

Pokud máme návrh obrysu, popřípadě skutečných součástí hlavice hotov, vrátíme se do menu HLAVICE a v rámci submenu ZADÁNÍ vybereme položku TĚLESA. Zobrazí se menu pro definice skutečných těles. Pomocí položky DEFINICE postupně přivlastníme konkrétní tělesa (čáry kresby) jednotlivým úsekům základního kinematického schématu. Pokud bychom se během definice zmýlili, provedeme pomocí další položky v menu ZRUŠENÍ vybrané definice. Pokud máme provedeny všechny definice, přejdeme do menu ANIMACE a provedeme kontrolu skutečné hla-

vice animací v rozsahu zdvihu motoru, čímž odhalíme možné kolizní stavy, které by se v případě standardního postupu projevily teprve při sestavení skutečné hmotné hlavice. Takto můžeme provádět všechny úkony se skutečným obrazem úchopné hlavice, které jsme prováděli v předchozích krocích návrhu se schematem.

Tímto je návrh úchopné hlavice hotov a my můžeme přikročit k rozkreslování jednotlivých výrobních výkresů. Pro optimalizaci práce můžeme „rozebrat“ jednotlivé součásti hlavice, které jsou již v reálném měřítku. Pozor po provedené definici jednotlivých částí hlavice jsou tyto součástky sdruženy do bloků. Tyto bloky lze zrušit a získat tak původní entity kresby (čáry, oblouky...) pomocí položky ZRUŠVŠE v menu TĚLESA.

4.4.2 Postup při kontrole již navržené hlavice

Předpokládejme jistou znalost práce s návrhovým systémem v rozsahu minulé kapitoly.

V mnoha případech stojí konstruktér před problémem pouze upravit již navrženou, třeba i vyrobenou či právě nevyužívanou úchopnou hlavici, aby byla schopná splňovat poněkud odlišné parametry, než na které byla navržena a současně přitom ponechat maximum součástí původních. Právě v takových případech by tento návrhový systém našel uplatnění pro svoji pružnost a právě proto, že je pouhou nadstavbou či nástrojem jiného většího CAD systému, jakým je právě AutoCAD. Budeme se tedy zabývat řešením tohoto ukázkového problému.

Spustíme AutoCAD s výkresem původní hlavice. Umístíme úchopnou hlavici tak, aby její podélná osa splynula s osou globálního souřadného systému a přistoupíme k zadávání kinematického schematu. V submenu SCHEMA vybereme patřičný typ modifikace struktury úchopné hlavice a typ úchopných prvků. Zvolíme přibližně podobné rozměry, jaké má skutečná hlavice. Dále přikročíme k přizpůsobení schematu této hlavici pomocí nástrojů k modifikaci umístění jednotlivých uzlových bodů, protažení či rotaci částí schematu, přičemž je důležitá pouze horní část kinematického schematu, spodní se dokreslí automaticky.

Pokud souhlasí horní část schematu se skutečnou hlavicí, přistoupíme k definici jednotlivých součástí, kterou provedeme již uvedeným způsobem.

Dalším krokem bude zadání parametrů motoru. Obdobně jako v předchozím případě budeme nejprve definovat dolní a horní úvrať motoru, přičemž tyto polohy odhadneme podle chování skutečných dílů hlavice. Provedeme návrh motoru tak, že zadáme skutečný existující průměr válce motoru a navrhovaný tlak tekutiny.

Opět známým způsobem zadáme typické parametry manipulovaného objektu (pro původně navrženou hlavicí) a rozměry čepů.

V tomto okamžiku můžeme provést kinematickou a pevnostní kontrolu skutečné hlavice podle postupů uvedených v minulé kapitole.

Naším úkolem je však modifikovat hlavicí tak, aby splňovala odlišné parametry. Provedeme tedy nové zadání objektu manipulace, tentokrát podle nových požadavků a nastavíme v menu SCHEMA patřičné parametry minimálního, maximálního a optimálního (požadovaného) uchopovaného průměru. Nyní můžeme pomocí nástrojů v menu SCHEMA modifikovat jednotlivé součásti skutečné hlavice včetně kinematického schematu. Budeme se snažit o co nejmenší nároky na úpravy stávající hlavice, aby byl problém řešen například pomocí výměny jednoho dílu, popřípadě jiným seřízením.

Po provedené modifikaci přistoupíme ke kinematické a pevnostní kontrole, přičemž (pokud hlavice vyhovuje) necháváme vykreslovat grafy důležitých závislostí a průběžně po každé modifikaci mechanismu a provedené pevnostní kontrole porovnávané získané grafy, čímž získáme podrobné podklady pro rozhodování o dalších modifikacích.

Tímto postupem získáme sestavný výkres reálné úchopné hlavice, přičemž dále buď rozkresluje pouze ty části, které byly modifikovány, nebo přímo provedeme export součástí prostřednictvím nástrojů z menu POMŮCKY pro další zpracování v systému pro generování NC programů. Takto lze například prostřednictvím nástroje EXP.DXF provést export dat o součásti pro systém APS a odtud přímo k obráběcímu stroji.

4.4.3 Podrobný popis menu HLAVICE

Obrazovkové menu HLAVICE:

AutoCAD - návrat do menu AutoCAD

Zadání - podmenu pro zadávání jednotlivých prvků hlavice

Kontrola - podmenu pro kinematickou a pevnostní kontrolu

Pomůcky - podmenu s pomocnými nástroji

Ulož - uložení současného stavu výkresu a důležitých hodnot proměnných do souboru

Načti - načtení obrazu a hodnot proměnných (starší varianty) ze souboru

Ukonči - ukončení práce v návrhovém systému HLAVICE, uložení obrazu a dat do souboru a ukončení AutoCADu

Odstraň - odstranění všech proměnných a pomocných procedur z paměti - uvolnění oblasti pro další aplikace v AutoLISPu

Menu ZADÁNÍ:

Hřídél - Zadání rozměrů uchopovaného objektu a výpočet potřebné úchopné síly

Schema - Submenu pro zadání kinematického schematu úchopné hlavice

Motor - Submenu pro návrh parametrů a výpočet motoru

Tělesa - Submenu pro definice skutečných součástí úchopné hlavice

Čepy - Zadání parametrů čepů

Menu SCHEMA:

VyberTyp - Výběr typu základního schematu a úchopných prvků

PosuňBod - Modifikace polohy uzlového bodu základního schematu

Protáhni - Protažení či zkrácení části schematu

Otoč - Otočení části schematu

Zrcadli - Dokreslení spodní části schematu podle horní

Měřítko - Zmenšení či zvětšení části schematu

Zpět - Zrušení posledního provedeného kroku

Změna:

Dmin - Modifikace minimálního uchopovaného průměru

Dmax - Modifikace maximálního uchopovaného průměru

Dopt - Modifikace optimálního uchopovaného průměru

Menu MOTOR:

Hor.úvr. - Označit nynější stav jako horní úvrať motoru

Dol.úvr. - Označit nynější stav jako dolní úvrať motoru

Cel na PR - Sevrít čelisti na zadaný průměr

PosunMot - Posunout motor na zadaný zdvih

AnimMot - Animace podle posunu motoru v závislosti na pohybu grafického kurzoru

AnimČel - Animace podle rozevření čelistí v závislosti na pohybu grafického kurzoru

MinPrům - Sevřít čelisti na minimální úchopný průměr

MaxPrům - Sevřít čelisti na maximální úchopný průměr

OptPrům - Sevřít čelisti na optimální úchopný průměr

NÁVRH - Submenu pro návrh rozměrů motoru a síly na motoru

Menu NÁVRH:

ZadSíly - Návrh rozměrů motoru podle zadané síly na motoru a vykreslení schématu motoru

ZadPrům - Výpočet síly na motoru podle zadaného průměru válce s vykreslením schématu motoru

Menu TĚLESA:

Definice - Definice dílčí součásti skutečné úchopné hlavice s přiřazením k příslušné části základního kinematického schématu

Zrušení - Zrušení definice jednotlivé součásti

ZrušVše - Zrušení definice všech definovaných součástí

Menu KONTROLA:

Grafy - Submenu pro nastavení parametrů pro vykreslování grafů

Animace - Submenu pro kinematickou kontrolu úchopné hlavice

Kontrola - Spuštění pevnostní kontroly úchopné hlavice a automatickou animací podle zdvihu motoru s vykreslením vybraných grafů

Pomůcky - Submenu s ostatními pomocnými nástroji

Menu GRAFY:

Které - Nastavení, které grafy se mají vykreslovat

Styl - Výběr typu podkladu grafů

Interpol - Výběr typu interpolace vykreslovaných grafů

Tloušťka - Zadání tloušťky křivek grafů

UprGraf - Změna typu interpolace vybrané již vykreslené křivky

Menu ANIMACE:

AnimMot - Animace podle posunu motoru v závislosti na pohybu grafického kurzoru

AnimČel - Animace podle rozevření čelistí v závislosti na pohybu grafického kurzoru

Cel na PR - Sevřít čelisti na zadaný průměr

PosunMot - Posunout motor na zadaný zdvih

MinPrům - Sevřít čelisti na minimální úchopný průměr

MaxPrům - Sevřít čelisti na maximální úchopný průměr

OptPrům - Sevřít čelisti na optimální úchopný průměr

MinMotor - Posunout motor na nulový zdvih

MaxMotor - Posunout motor na maximální zdvih

Menu POMŮCKY:

Zobraz - Submenu pro nastavení pohledu zobrazení

UprGraf - Změna typu interpolace vybrané již vykreslené křivky

Nastav - Submenu nastavení parametrů programu

Ulož - uložení současného stavu výkresu a hodnot proměnných do souborů

Načti - načtení obrazu a hodnot proměnných (starší varianty) ze souboru

TISK GR - Nastavení tisku grafu podle výběru

PLOT GR - Nastavení vykreslování grafu podle výběru

Proved' - Vykreslení/tisk podle nastavení

EXP.DXF - Export výkresu (bez grafů) ve formátu DXF

EXP.DWG - Export výkresu (bez grafů) ve formátu DWG

IMP.DXF - Import výkresu ve formátu DXF

IMP.DWG - Import výkresu ve formátu DWG

Menu ZOBRAZ:

Pohled - Nastavení typu pohledu podle výběru

Graf - Zobrazení grafu podle výběru do aktuálního grafického okna

Menu NASTAV:

CestaDWG - Modifikace cesty do adresáře pro ukládání obrazů při povelu ULOŽ

CestaHLV - Modifikace cesty do adresáře pro ukládání dat při povelu ULOŽ

Roletové menu ZOBRAZ:

Varianta0 - Zobrazí pouze grafy varianty 0

Varianta1 - Zobrazí pouze grafy varianty 1

... ..

Varianta9 - Zobrazí pouze grafy varianty 9

Překryj - Submenu pro překrytí právě zobrazených grafů dalšími vybranými grafy

Motor - Zobrazí schema motoru

Schema - Zobrazí kinematické schema úchopné hlavice

Úchop pr. - Zobrazí kružnici uchopovaného průměru

Síť grafů - Zobrazí podkladovou síť grafů s desetinným dělením

Roletové menu PŘEKRYJ:

Varianta0 - Zobrazí graf varianty 0 k již existujícím zobrazeným variantám

Varianta1 - Zobrazí graf varianty 1 k již existujícím zobrazeným variantám

... ..

Varianta9 - Zobrazí graf varianty 9 k již existujícím zobrazeným variantám

Zobraz(zpět) - Návrat do submenu ZOBRAZ

Roletové menu SKRYJ/ZRUŠ:

Motor - Vypne zobrazování schematu motoru

Schema - Vypne zobrazování kinematického schematu úchopné hlavice

Úchop pr. - Vypne zobrazování kružnice uchopovaného průměru

Síť grafu - Vypne zobrazování podkladové sítě grafů s desetinným dělením

ZRUŠ GRAFY - Vymaže (zruší) všechny varianty grafů

5 Příklad automatizovaného návrhu MÚH

5.1 Zadání požadované hlavice

Máme za úkol navrhnout kinematické schema mechanické úchopné hlavice s kruhovými úchopnými prvky a pneumatickým přímočarým motorem pro zadaný manipulovaný objekt [Příloha č.7] pro průmyslový robot KUKA IR 161/60, minimální uchopovaný průměr 46mm a maximální uchopovaný průměr 70mm.

5.2 Příklad řešení automatizovaného návrhu hlavice

Pomocí výše uvedeného postupu byl proveden výpočet potřebné úchopné síly a středu uchopení manipulovaného objektu s těmito výsledky:

$$F_u = 760 \text{ N}$$

$$X_u = 60 \text{ mm}$$

$$m = 2.65 \text{ kg}$$

Bylo navrženo kinematické schema transformačního bloku a pneumatický přímočarý motor [Příloha č.8]. Byla úspěšně provedena kinematická i pevnostní kontrola. Výsledky kontroly jsou uvedeny v grafech důležitých závislostí [Příloha č.9].

Pro ukázkou, jak se změní některé parametry hlavice bylo provedeno vyhodnocení pevnostní kontroly pro upravené schema o nepatrný posun bodu 4 [Příloha č.11]. Výsledky jsou uvedeny přehledně v grafech některých závislostí [Příloha č.12].

Pro ukázkou, jak se změní některé důležité závislosti při změně typu úchopných prvků bylo modifikováno opět původní schema [Příloha č.13] při zachování všech ostatních parametrů. Výsledky jsou uvedeny v grafech některých důležitých závislostí [Příloha č.14].

6 Závěr

V této práci je ukázán možný směr vývoje návrhových systémů ve strojírenství. Jasně z výše uvedených důvodů vyplývá, že se musí navrhovaný návrhový systém buď implementovat do existujícího obecného CAD systému nebo musí být zajištěno jeho propojení s některým z těchto CAD systémů, neboť budoucnost výpočetní techniky jasně směřuje k větší datové i systémové kompatibilitě. Pokud se týče tohoto konkrétního návrhového systému pro návrh mechanické úchopné hlavice, jsou tyto požadavky bezezbytku splněny.

Během postupného vývoje se však výrazně projevila malá výpočetní rychlost použitých prostředků (PC-AT 286—AutoCAD—AutoLISP). Návrhový systém plní všechny požadavky, co se týče požadovaných úkonů a činností. Avšak v případě, že pracujeme již delší dobu v grafickém editoru a plně využíváme možností úschovy různých variant grafů kreslených na větší počet iterací, klesá rychlost AutoCADu natolik, že konstruktér ztratí trpělivost na další optimalizace již vyhovujícího řešení. Situace se ještě více vyhrotí v případě, že do editoru načítáme větší množství obrazců skutečných součástí úchopných hlavic.

Tyto poznatky plynou z vývoje návrhového systému na osobním počítači AT 286/25 bez matematického koprocessoru s 1MB operační paměti, což není pro grafické aplikace nejvýhodnější konfigurace. Situace se poněkud zlepší při použití výkonnějšího počítače, ale ani tehdy nelze provádět takové činnosti jako je například animace mechanismu v reálném čase. Zajímavým tématem dalšího výzkumu v této oblasti by mohla být varianta s využitím ADS namísto AutoLISPu.

Z těchto výsledků vyplývá, že prostředí osobních počítačů není určeno pro náročné grafické a výpočetní návrhové systémy operující s velkým množstvím entit. Prozatímni použití těchto počítačů pro řešení dílčích problémů, jakým je například řešení návrhu MÚH, možné je.

A Seznam použité literatury

- [1] Matička,R., Talácko,J.: Mechanismy manipulátorů a průmyslových robotů.
SNTL 1991
- [2] Chvála,B., Matička,R., Talácko,J.: Průmyslové roboty a manipulátory.
SNTL 1990
- [3] Matička,R., Talácko,J.: Konstrukce průmyslových robotů.
ČVUT Praha 1989
- [4] Fiala,F.: Diplomová práce. VŠST Liberec 1992
- [5] AutoLISP – Programátorská příručka. Autodesk 1991
- [6] Vývojový systém AutoCADu – Programátorská příručka. Autodesk 1991
- [7] Vosáhlo,M.: Grafické standardy a přenosy dat mezi systémy CAD.
PC-World 11/91
- [8] Hallidayová,C.: Špičkový CAD. PC-World 11/91
- [9] Satrapa,P.: $\text{\LaTeX}$, přehled příkazů. VŠST Liberec 1991

B Seznam příloh

1. Disketa s návrhovým systémem Hlavice
2. Zásadní část definičního souboru menu Hlavice
3. Komentované výpisy některých zdrojových souborů
4. Celkové rozložení kreslících ploch
5. Ukázka druhů grafů
6. Ukázka interpolací grafů
7. Schema manipulovaného objektu pro řešený příklad
8. Výsledné kinematické schema řešeného příkladu
9. Grafy důležitých závislostí pro řešený příklad
10. Jednotlivé fáze pohybu kinematického schematu řešeného příkladu
11. Schema první alternativní varianty
12. Grafy důležitých závislostí pro první alternativní variantu
13. Schema druhé alternativní varianty
14. Grafy důležitých závislostí pro druhou alternativní variantu

Příloha č. 2

Zásadní část definičního souboru menu HLAVICE.MNU

***POP8

**popzobr

[Zobraz]

[Varianta0]^P(vari 0)

[Varianta1]^P(vari 1)

[Varianta2]^P(vari 2)

[Varianta3]^P(vari 3)

[Varianta4]^P(vari 4)

[Varianta5]^P(vari 5)

[Varianta6]^P(vari 6)

[Varianta7]^P(vari 7)

[Varianta8]^P(vari 8)

[Varianta9]^P(vari 9)

[Prekryj]^P8=prekryj \$P8=*

[--]

[Motor]^Player on hlschpom

[Schema]^Player on hlschema

[Uchop pr.]^Player on hluchpru

[Sit grafu]^Player on hlmrizov

**prekryj

[Prekryj]

[Varianta0]^Player on hlgr0 \$P8=popzobr

[Varianta1]^Player on hlgr1 \$P8=popzobr

[Varianta2]^Player on hlgr2 \$P8=popzobr

[Varianta3]^Player on hlgr3 \$P8=popzobr

[Varianta4]^Player on hlgr4 \$P8=popzobr

[Varianta5]^Player on hlgr5 \$P8=popzobr

[Varianta6]^Player on hlgr6 \$P8=popzobr

[Varianta7]^Player on hlgr7 \$P8=popzobr

[Varianta8]^Player on hlgr8 \$P8=popzobr

[Varianta9]^Player on hlgr9 \$P8=popzobr

[--]

[Zobraz (zpet)]^P8=popzobr \$P8=*

***POP9

**xxxx

[Skryj/Zrus]

[Motor]^Player off hlschpom ;

[Schema]^Player off hlschema ;

[Uchop pr.]^Player off hluchpru ;

[Sit grafu]^Player off hlmrizov ;

[--]

[^ZRUS GRAFY]^P(load "z24")

***icon

**schemata

[TYP KINEMATICKÉHO SCHEMATU]

[hlavice(s01)]^P(setq vybr 1)

[hlavice(s02)]^P(setq vybr 2)

[hlavice(s03)]^P(setq vybr 3)

**celisti

[TYP UCHOPNYCH PRVKU]

[hlavice(c01)]^P(setq typcel 1)

[hlavice(c02)]^P(setq typcel 2)

**pohledy

```
[hlavice(pohl4)]^Pvports r pohl4
[hlavice(pohl5)]^Pvports r pohl5
[hlavice(pohl6)]^Pvports r pohl6
```

**mrize

```
[Podklad grafu]
[hlavice(mrizene)]^PPlayer off hlmrizov off hlobrub ;
[hlavice(mrizejo)]^PPlayer on hlmrizov off hlobrub ;
[hlavice(obruba)]^PPlayer on hlobrub off hlmrizov ;
```

**uprgraf

```
[Typ interpolace]
[hlavice(int1)]^P(upkr "d")
[hlavice(int2)]^P(upkr "f")
[hlavice(int3)]^P(upkr "s")
```

**interpol

```
[Typ interpolace]
[hlavice(int1)]^P(setq interp "d")
[hlavice(int2)]^P(setq interp "f")
[hlavice(int3)]^P(setq interp "s")
```

**graf1

```
[Tisknout/Kreslit graf:]
[hlavice(graf1)]^P(setq vybgr 1)
[hlavice(graf2)]^P(setq vybgr 2)
[hlavice(graf3)]^P(setq vybgr 3)
[hlavice(graf4)]^P(setq vybgr 4)
[hlavice(graf5)]^P(setq vybgr 5)
[hlavice(graf6)]^P(setq vybgr 6)
[hlavice(graf7)]^P(setq vybgr 7)
[hlavice(graf8)]^P(setq vybgr 8)
[hlavice(graf9)]^P(setq vybgr 9)
[hlavice(graf10)]^P(setq vybgr 10)
```

**graf2

```
[Zobrazit:]
[hlavice(graf1)]^P(zobrgraf 1)
[hlavice(graf2)]^P(zobrgraf 2)
[hlavice(graf3)]^P(zobrgraf 3)
[hlavice(graf4)]^P(zobrgraf 4)
[hlavice(graf5)]^P(zobrgraf 5)
[hlavice(graf6)]^P(zobrgraf 6)
[hlavice(graf7)]^P(zobrgraf 7)
[hlavice(graf8)]^P(zobrgraf 8)
[hlavice(graf9)]^P(zobrgraf 9)
[hlavice(graf10)]^P(zobrgraf 10)
[hlavice(s01)]^Pzoom w -220,-170 600,390
[hlavice(hodnoty)]^Pzoom w 980,-170 1170,-50
```

***SCREEN

**hlavice

```
[AutoCAD]^C^Cmenu acad
[* * * *]^S=OSNAPB
[Zadani]^P$s=zadani $s=*
[Kontrola]^S=kontrola $s=*
[Pomucky]^P(load "z18") $s=pomucky $s=*
[]
[Uloz]^P(load "z18")(ulozdata)(nez18)
[Nacti]^P(load "z18")(nactdata)(nez18)
[]
[]
[]
[]
[]
[]
[]
[]
[]
```

```
[Okonci]^Pscript konec
[Odstran]^P(if hlavice (load "z31"))
[]
[Pomoc]^P(hlhelp 1)
[]
```

**zadani

```
[Hridel]^P(nez2)(load "z1")(load "z2") $s=*
[Schema]^P(load "z3") $s=schema $s=*
[Motor]^P(setq memotor T)(load "z6")$s=motor $s=*
[Telesa]^P(load "z7")$s=telesa $s=*
[Cepy]^P(load "z27")
[]
[]
[]
[]
[]
[]
[]
[]
[]
[navrat]^P$s=
[]
[]
[]
[]
[Pomoc]^P(hlhelp 2)
[]
```

**schema

```
[VyberTyp]^P($i=schemata $i=* $i=celisti $i=* $s=* ^P(load "z2")(load "z5") $s=
=schema $s=*
[PosunBod]^P(posunbod)
[Protahni]^Pstretch c
[Dtoc]^Protate
[Zrcadli]^P(nacti)(kresan)
[Meritko]^Pscale c
[Zpet]^Pu
[]
[Zmena:]
[Dmin]^P(dejhod "Minimalni uchopovany prumer Dmin" (DMIN)
[Dmax]^P(dejhod "Maximalni uchopovany prumer Dmax" (DMAX)
[Dopt]^P(dejhod "Pozadovany (optimalni) uchopovany prumer Dopt" (DOPT)
[]
[]
[navrat]^P(nez3) $s=
[]
[]
[]
[]
[Pomoc]^P(hlhelp 3)
[]
```

**motor

```
[Hor.uvr.]^P(if (< X1 XOD) (setq XOH XOD)(setq XOH X1))
[Dol.uvr.]^P(if (> X1 XOH) (setq XOD XOH)(setq XOD X1))
[Cel na PR]^P(celnapr) $s=*
[PosunMot]^P(posunmot) $s=*
[AnimMot]^P(animmot) $s=*
[AnimCel]^P(animcel) $s=*
[MinPrum]^P(minprum)
[MaxPrum]^P(maxprum)
[OptPrum]^P(optimal)
[]
[NAVRH]^P$s=navrh $s=*
[]
[]
[navrat]^P(nez6)(setq memotor nil)$s=
[]
[]
[]
[]
```

```
[[
[Pomoc]^P(hlhelp 4)
[[
```

[illegible]

```

**telesa
[Definice]^P(telesa) $$$=
[Zruseni]^P(rozbite1) $$$=
[ZrusVse]^P(rozbivse) $$$=

```

```
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[navrat]^P(nez7)$s=  
[ ]  
[ ]  
[ ]  
[ ]  
[Pomoc]^P(hlhelp_5)  
[ ]
```

```

**kontrola
[Grafy]$s=grafy $$=*
[Animace]^P(load "z6")$$=animace $$=*
[Kontrola]^P(load "z21")

```

```
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[ ]  
[navrat]$s=  
[ ]  
[ ]  
[Pomucky]^P(load "z18") $$=pomucky $$=*  
[ ]  
[Pomoc]^P(hlhelp ?)  
[ ]
```

```

**grafy
[Ktere]^P(load "z1/")
[Styl]^P($i=mrize $i=*)
[Interpol]^P($i=interpol $i=*)
[Tloustka]^P(dejhod "Sila cary grafu" tlgraf)
[]
[]
[UprGraf]^P($i=uprgraf $i=*)
[]
[]
[]
[]
[]
[]
[]
[]
[navrat]^P($s=)
[]
[]
[]
[]
[Pomoc]^P(hlhelp 13)
[]

```

```

**animace
[AnimMot]^P(animmot)
[AnimCel]^P(animcel)
[Cel na PR]^P(celnapr)
[PosunMot]^P(posunmot)
[MinPrum]^P(minprum)
[MaxPrum]^P(maxprum)
[OptPrum]^P(optimal)
[MinMotor]^P(minmot)
[MaxMotor]^P(maxmot)
[]
[]
[]
[]
[]
[navrat]^P(nez6) $s=
[]
[]
[]
[]
[Pomoc]^P(hlhelp 8)
[]

```

```

**pomucky
[Zobraz]^P($s=zobraz $s=*)
[UprGraf]^P($i=uprgraf $i=*)
[Nastav]^P($s=nastav $s=*)
[Uloz]^P(ulozdata)
[Nacti]^P(nactdata)
[]
[TISK GR]^P($i=graf1 $i=*) (tiskgr vybgr 0)
[PLOT GR]^P($i=graf1 $i=*) (tiskgr vybgr 1)
[Proved]^P(pscript hlavprac)
[EXP.DXF]^P(pdxfout)
[EXP.DWG]^P(progn (setq naz (getstring "Nazev souboru: ")) (command "block"
naz "0,0" "w" "-220,-170" "600,390" "" "wblock" naz naz "insert" naz "0,0" "" "
" "" "explode" (entlast)))
[INP.DXF]^P(pdxfin)
[IMP.DWG]^P(pinsert)
[]
[navrat]^P(nez18)$s=
[]
[]
[]
[]
[]
[Pomoc]^P(hlhelp 9)
[]

```


Příloha č. 3.

Komentované výpisy některých zdrojových souborů

(Z2.LSP, Z8.LSP, Z21.LSP, Z29.LSP
a Z30.LSP)

Z2.LSP

```
(defun NEZ2() : zruseni vseh funkci ze z2(3).lsp a z8(9,10).lsp
  (setq vypmot nil
        vypcel nil
        kresan nil
        nacti nil
        kresancel nil
        naccel nil
        vypcelm nil
        vypcelpr nil
        nez2 nil
  )
)
```

```
(defun NACT1(/ ZNAM ed1 ed2 ed3 ed4 ed5 ed1s ed2s ed3s ed4s ed5s en t1 GAMA
```

```
1 )
(if (= en1 nil) : prvni nacteni entit zakladniho schematu
  (progn
    (setq en (entnext))
    (while en : najit prvni caru schematu
      (if (= "HLSCHEMA" (cdr (assoc 8 (entget en))))
        (setq enp en en nil) : prvni cara hladiny "hlschema"
        (setq en (entnext en)) : dalsi prvek
      )
    )
    (setq en1 enp)
    (mapcar (lambda(a1) : nacist entity zakladniho schematu
              (set a1 (entnext enp))
              (setq enp (eval a1))
            )
            (en2 en3 en4 en5 en1s en2s en3s en4s en5s)
          )
    (if (entnext enp)
      (progn
        (prnaccel) : nacteni entit existujicich celisti
        (setq en (entnext))
        (while en : najit kruznici uchopov. objektu
          (if (= "HLUCHPRU" (cdr (assoc 8 (entget en))))
            (setq enk en en nil) : to je vona
            (setq en (entnext en)) : dalsi prvek
          )
        )
      )
    )
  )
)

(mapcar (lambda (a1 a2) : nacteni dat entit zakl. sch.
  (set a1 (entget (eval a2)))
)
(ed1 ed2 ed3 ed4 ed5 ed1s ed2s ed3s ed4s ed5s)
(en1 en2 en3 en4 en5 en1s en2s en3s en4s en5s)
)
```

odu (mapcar (lambda (a1 a2 a3 a4) : nastaveni promennych souradnic uzl. b

```
(set a1 (car (cdr (assoc (eval a3) (eval a4)))))
(set a2 (car (cdr (cdr (assoc (eval a3) (eval a4)))))
)
(X1 X2 X3 X4 X5 X6)
(Y1 Y2 Y3 Y4 Y5 Y6)
(10 10 10 10 10 11)
(ed1 ed2 ed3 ed4 ed5 ed5)
```

2

```
(mapcar (lambda (l a1 a2 a3 a4) : vypocet delek prvku
(set l (sqrt (+ (expt (- (eval a1) (eval a2)) 2) (expt (- (eval a3)
(eval a4)) 2)))))
```

```
(L1 L2 L3 L4 L5)
(x2 x3 x4 x5 x6)
(x1 x2 x3 x4 x5)
(y2 y3 y4 y5 y6)
(y1 y2 y3 y4 y5)
```

```
(mapcar (lambda (a1 a2 a3 a4 a5 a6) : vypocet uhlu alfa1-alfa5
(progn
(if (< (eval a1) (eval a2)) (setq ZNAM -1)
(setq ZNAM 1)
)
(if (= (eval a3) (eval a4)) (set a5 (* ZNAM (* 3.1415 0.5
)))
(set a5 (* ZNAM (acos (/ (- (eval a3) (eval a
4)) (eval a6)))))
)
)
)
(Y2 Y3 Y4 Y5 Y6)
(Y1 Y2 Y3 Y4 Y5)
(X2 X3 X4 X5 X6)
(X1 X2 X3 X4 X5)
(ALFA1 ALFA2 ALFA3 ALFA4 ALFA5)
(L1 L2 L3 L4 L5)
```

```
(setq BETA (- ALFA5 ALFA4))
(setq T1 (sqrt (+ (expt (- X3 X5) 2) (expt (- Y3 Y5) 2))))
(if (< Y5 Y3) (setq Z1 -1)
(setq Z1 1)
)
(if (> X3 X4) (setq Z3 -1)
(setq Z3 1)
)
(setq GAMA1 (* (acos (/ (- X5 X3) T1)) Z1))
(setq Z2 (/ (+ Y4 (- (* (sin GAMA1) (- X5 X4)) Y5)) (sqrt (expt (+ Y4 (- (*
(sin GAMA1) (- X5 X4)) Y5)) 2))))
```

```
(if (= nenaccl nil) (naccl)) : nacteni celisti
```

```
) : konec NACT1()
```

```
(defun PRNACCEL() : prvni nacteni celisti
(setq en6 (entnext enp)) : vrchni celist
(setq en7 (entnext en6))
(setq enp (setq en8 (entnext en7)))
(if (= typcel 2) :
(setq enp (setq en9 (entnext en8)))
)
(setq en6s (entnext enp)) : spodni celist
(setq en7s (entnext en6s))
(setq enp (setq en8s (entnext en7s)))
(if (= typcel 2)
(setq enp (setq en9s (entnext en8s)))
)
```

```
) ; konec prnacel()
```

```
(defun KRESAN(/ edk) ; nakresleni skutecneho schematu pri animaci
```

```
(setq D (* 2 R0))
```

```
; test uchopovaneho prumeru
```

```
(if (< D (- DMIN 0.1)) (prompt "\nPozor, uchopovany prumer je mensi nez zadana minimalni mez")
```

```
(if (> D (+ DMAX 0.1)) (prompt "\nPozor, uchopovany prumer je vetsi nez zadana maximalni mez")
```

```
(if (and (> D (- DOPT 0.1)) (< D (+ DOPT 0.1))) (prompt "\nOptimalni uchopovany prumer")
```

```
(prompt "\n")
```

```
)
```

```
)
```

```
)
```

```
(mapcar (lambda (a1 a2 a3 a4 a5 / ed)
```

```
(setq ed (entget (eval a1)))
```

```
(setq ed (subst (cons 10 (cons (eval a2) (cons (eval a3) nil)))
```

```
(assoc 10 ed) ed))
```

```
(setq ed (subst (cons 11 (cons (eval a4) (cons (eval a5) nil)))
```

```
(assoc 11 ed) ed))
```

```
(entmod ed)
```

```
)
```

```
(en1 en2 en3 en4 en5 en1s en2s en3s en4s en5s)
```

```
(X1 X2 X3 X4 X5 X1 X2 X3 X4 X5)
```

```
(Y1 Y2 Y3 Y4 Y5 (* -1 Y1) (* -1 Y2) (* -1 Y3) (* -1 Y4) (* -1 Y5))
```

```
(X2 X3 X4 X5 X6 X2 X3 X4 X5 X6)
```

```
(Y2 Y3 Y4 Y5 Y6 (* -1 Y2) (* -1 Y3) (* -1 Y4) (* -1 Y5) (* -1 Y6))
```

```
(kresancel) ; nakresleni celisti
```

```
; kresleni kruznice
```

```
(setq edk (entget enk))
```

```
(setq edk (subst (cons 40 R0) (assoc 40 edk) edk)) ; prumer
```

```
(setq edk (subst (cons 10 (cons x11 (cons y11 nil))) (assoc 10 edk) edk))
```

```
; stred
```

```
(entmod edk)
```

```
; kresleni pistu
```

```
(if enpist
```

```
(progn
```

```
(setq edk (entget enpist))
```

```
(setq edk (subst (cons 10 (cons X1 (cons Y1 nil))) (assoc 10 edk) edk))
```

```
(entmod edk)
```

```
)
```

```
)
```

```
(mapcar (lambda (a2 a3 a4 a5 a6 a7 / ed)
```

```
(if (/= (eval a2) nil)
```

```
(progn
```

```
(setq ed (entget (eval a2)))
```

```
(setq ed (subst (cons 10 (cons (eval a3) (cons (eval a4) nil))) (assoc 10 ed) ed))
```

```
(setq ed (subst (cons 50 (- (angle (list (eval a3) (eval a4))) (list (eval a5) (eval a6))) (eval a7))) (assoc 50 ed) ed))
```

```
(entmod ed)
```

```
)
```

```
)
```

```
)
```

```
(enb1 enb2 enb3 enb4 enb5 enb1s enb2s enb3s enb4s enb5s)
```

```
(X1 X2 X3 X4 X5 X1 X2 X3 X4 X5)
```

```
(Y1 Y2 Y3 Y4 Y5 (* -1 Y1) (* -1 Y2) (* -1 Y3) (* -1 Y4) (* -1 Y5))
```

```
(X2 X3 X4 X5 X6 X2 X3 X4 X5 X6)
```

```
(Y2 Y3 Y4 Y5 Y6 (* -1 Y2) (* -1 Y3) (* -1 Y4) (* -1 Y5) (* -1 Y6))
```

```
(X3 X4 X5 X6 X7 X3 X4 X5 X6 X7)
```

```
(Y3 Y4 Y5 Y6 Y7 (* -1 Y3) (* -1 Y4) (* -1 Y5) (* -1 Y6) (* -1 Y7))
```

```

X6      )
      (Y2 Y3 Y4 Y5 Y6 (* -1 Y2) (* -1 Y3) (* -1 Y4) (* -1
Y5)      (* -1 Y6) )
      (ZUB1 ZUB2 ZUB3 ZUB4 ZUB5 ZUB1S ZUB2S ZUB3S ZUB4S
ZUB5S )
)
)      : konec KRESAN()

```

```

(defun VYPMOT(/ T1 GAMA1 GAMA2 E F G kousek prac) : vypocet bodu pro a
animaci podle X

```

```

: Podminky necelistivosti:

```

```

(if (= memotor nil) : test na roztrhání motoru
(progn
  (if (< x 0) (setq x 0))
  (if (> x (- XOH XOD)) (setq x (- XOH XOD)))
)
)

```

```

(if (> (- (expt (- L4 L3) 2) (expt (- Y5 Y3) 2)) 0)
  (setq E (sqrt (- (expt (- L4 L3) 2) (expt (- Y5 Y3) 2))))
)

```

```

(setq F (- (- X5 L1) (* L2 (cos ALFA2))))
(setq G (sqrt (- (expt (+ L4 L3) 2) (expt (- Y5 Y3) 2))))

```

```

(setq kousek (/ (- G F) 500)) : nasilne posunuti do rozumnych mezi

```

```

(cond ((<= x (- F G)) (setq X (+ (- F G) kousek)))
      ((>= x (+ F G)) (setq X (- (+ F G) kousek)))
      ((/= E nil)
        (if (and (>= x (- F E))
                  (<= x (+ F E)))
            (if (< X F) (setq X (- (- F E) kousek)) (setq X (+ (+
F E) kousek)))
            )
        )
      )
)

```

```

(setq X1 (+ X XOD))
(setq X2 (+ X1 L1))
(setq X3 (+ X1 (+ L1 (* L2 (cos ALFA2)))))
(setq Y3 (* L2 (sin ALFA2)))

```

```

(setq T1 (sqrt (+ (* (- X3 X5) (- X3 X5)) (* (- Y3 Y5) (- Y3 Y5))))
(setq GAMA1 (* Z1 (acos (/ (- X5 X3) T1))))
(setq prac (/ (- (* L3 L3) (+ (* L4 L4) (* T1 T1))) (* (* -2 L4) T1)))
(setq GAMA2 (* Z2 (acos prac)))
(setq X4 (- X5 (* L4 (cos (- GAMA2 GAMA1)))))
(setq Y4 (+ Y5 (* L4 (sin (- GAMA2 GAMA1)))))

```

```

(setq ALFA4 (- GAMA1 GAMA2))
(setq ALFA5 (+ BETA ALFA4))

```

```

(setq X6 (+ X5 (* L5 (cos ALFA5))))
(setq Y6 (+ Y5 (* L5 (sin ALFA5))))

```

```

)      : konec VYPMOT()

```

```

(defun VYPCEL(/ B K) : vypocet bodu pro animaci podle celisti (X6,Y6)

```

```

: test na roztržení L3 :

```

```

(setq ALFA5 (asin (/ (- Y6 Y5) L5)))
(setq ALFA4 (- ALFA5 BETA))
(setq X4 (- X5 (* (cos (* -1 ALFA4)) L4)))
(setq Y4 (+ Y5 (* (sin (* -1 ALFA4)) L4)))

```

```

(setq X3 (- X4 (* Z3 (sqrt (- (* L3 L3) (expt (- Y4 Y3) 2))))))
(setq X2 (- X3 (* (cos ALFA2) L2)))
(setq X1 (- X2 L1))

(setq X (- X1 X0D))

(vypmot)

) ; konec VYPCEL()

```

; ***** nacteni NACCEL :

```

(cond ((= typcel 1)
      (load "z8")
    )
      ((= typcel 2)
      (load "z9")
    )
      ((= typcel 3)
      (load "z10")
    )
) ; konec cond

```

Z8.LSP

(defun VYPCELM() ; vypocet souradnic celisti podle X6, kruhove celisti

```

(setq ALFA6 (+ (- ALFA5 pi) BETA2))

(setq X6 (+ X5 (* (cos ALFA5) L5)))
(setq Y6 (+ Y5 (* (sin ALFA5) L5)))
(setq X7 (+ X6 (* (cos ALFA6) L6)))
(setq Y7 (+ Y6 (* (sin ALFA6) L6)))
(setq X8 (+ X7 (* (cos (- ALFA6 1.57)) L7)))
(setq Y8 (+ Y7 (* (sin (- ALFA6 1.57)) L7)))
(setq X9 (+ X8 (* (cos (- ALFA6 1.57)) R)))
(setq Y9 (+ Y8 (* (sin (- ALFA6 1.57)) R)))
(setq X11 (setq X10 X9))
(setq Y10 (+ Y9 R))
(setq Y11 0)
(setq R0 Y10)

```

) ; konec VYPCELM()

(defun VYPCELMR(/ A B C Y k znam rkrit) ; vypocet souradnic
; celisti podle prumeru D, kruhove celisti

; D = uchopovany prumer

```

(setq R0 (/ D 2))

```

```

(setq k (- pi BETA2))

```

```

(setq A (+ (expt (- (- (* L5 (sin k)) L7) R) 2) (expt (+ L6 (* L5 (cos k)))
2)))

```

```

(setq B (* (* 2 (- (- (* L5 (sin k)) L7) R)) (+ (- R R0) Y5)))

```

```

(setq C (- (expt (+ (- R R0) Y5) 2) (expt (+ L6 (* L5 (cos k))) 2)))

```

```

(setq rkrit (- (+ Y5 (* L5 (sin (- pi BETA2)))) L7))

```

```

(if (< r0 rkrit) (setq znam -1)

```

```

      (setq znam 1)

```

```

(setq ALFA6 (* znam (acos (/ (+ (* -1 B) (sqrt (- (* B B) (* (* 4 A) C))))
* 2 A))))))
(setq ALFA5 (- (+ pi ALFA6) BETA2))
(vypcelm)
) ; konec VYPCELPR()

(defun NACCEL(/ ed6 ed7 ed8 ed6s ed7s ed8s ZNAM) ; nacteni kruhových c
elisti
  (mapcar (lambda (a1 a2)
    (set a1 (entget (eval a2)))
  )
  (ed6 ed7 ed8 ed6s ed7s ed8s)
  (en6 en7 en8 en6s en7s en8s)
)

  (setq x7 (car (cdr (assoc 10 ed7))))
  (setq y7 (car (cdr (cdr (assoc 10 ed7)))))
  (setq x8 (car (cdr (assoc 11 ed7))))
  (setq y8 (car (cdr (cdr (assoc 11 ed7)))))
  (setq L6 (distance (list x6 y6) (list x7 y7)))
  (setq L7 (distance (list x7 y7) (list x8 y8)))

  (setq R (cdr (assoc 40 ed8)))
  (setq uhcel (- (cdr (assoc 51 ed8)) (cdr (assoc 50 ed8))))

  (if (< Y7 Y6) (setq ZNAM -1)
    (setq ZNAM 1)
  )
  (if (= X7 X6) (setq ALFA6 (* ZNAM (* 3.1415 0.5)))
    (setq ALFA6 (* ZNAM (acos (/ (- X7 X6) L6))))
  )
  (setq BETA2 (- (+ pi ALFA6) ALFA5))

) ; konec NACCEL()

(defun KRESANCEL(/ ed8 ed8s)

  (mapcar (lambda (a1 a2 a3 a4 a5 / ed)
    (setq ed (entget (eval a1)))
    (setq ed (subst (cons 10 (cons (eval a2) (cons (eval a3)
nil ))) (assoc 10 ed) ed))
    (setq ed (subst (cons 11 (cons (eval a4) (cons (eval a5)
nil ))) (assoc 11 ed) ed))
    (entmod ed)
  )
  (en6 en7 en6s en7s)
  (X6 X7 X6 X7)
  (Y6 Y7 (* -1 Y6) (* -1 Y7) )
  (X7 X8 X7 X8)
  (Y7 Y8 (* -1 Y7) (* -1 Y8) )
)

  (setq ed8 (entget en8)) (setq ed8s (entget en8s))
  (setq ed8 (subst (cons 10 (cons x9 (cons y9 nil ))) (assoc 10 ed8) ed
8))
  (setq ed8 (subst (cons 50 (- (+ ALFA6 1.57) (/ uhcel 2))) (assoc 50 e
d8) ed8))
  (setq ed8 (subst (cons 51 (+ (+ ALFA6 1.57) (/ uhcel 2))) (assoc 51 e
d8) ed8))
  (setq ed8s (subst (cons 10 (cons x9 (cons (* -1 y9) nil ))) (assoc 10
ed8s) ed8s))
  (setq ed8s (subst (cons 50 (- (- 4.71 ALFA6) (/ uhcel 2))) (assoc 50
ed8s) ed8s))
  (setq ed8s (subst (cons 51 (+ (- 4.71 ALFA6) (/ uhcel 2))) (assoc 51
ed8s) ed8s))

  (entmod ed8) (entmod ed8s)

) ; konec KRESANCEL()

```

```
*****
```

```
(defun GRAFMOT(/ en citac puvrstva seznam nazev)
  (if (null (and dopt dmin dmax x0d x0h) )
    (setq en (prompt "\nPostupujte prosim podle navodu !!!")
          graf nil
          vypocgr nil
          vztahx nil
          vztahy nil
          minent nil
          minbod nil
          grafmot nil
    )
  )
```

```
  (if (null varianta) (setq varianta 0) (setq varianta (if (= varianta 9) 0 (+ varianta 1))))
  (setq varianta (fix varianta))
  (vari varianta)
  (dejhod "Cislo varianty [0-9]" varianta)
  (if (or (< (setq varianta (fix varianta)) 0) (> varianta 9)) (setq varianta
0))
```

```
  (vari varianta)
  (if (or (null nazvar) (= "" nazvar) ) (setq nazvar (getvar "dwgname")))
  (prompt "\nNazev varianty : <")(princ nazvar)
  (setq nazev (getstring "> "))
  (if (/= nazev "") (setq nazvar nazev))
  (setq puvrstva (getvar "clayer"))
  (if (null barvar) (setq barvar 7))
  (dejhod "Barva varianty (1-7)" barvar)
```

```
  (vari varianta)
  (setq grvrstva (strcat "hlgr" (itoa (fix varianta))))
  (command "layer" "m" grvrstva "")
```

```
  (if (anone "Smazat puvodni obsah vrstvy" 1) (nolayer grvrstva))
```

```
  (nact1)
  (if (null kroku) (setq kroku 10))
  (if (null fuz) (setq fuz 0.0))
  (if (null fmz) (setq fmz 0.0))
  (if (null ftr) (setq ftr 0.1))
  (dejhod "Pocet kroku" kroku)
  (dejhod "Konstantni upinaci sila [N]" fuz)
  (dejhod "Konstantni sila na motoru [N]" fmz)
  (dejhod "Koefficient treni [1]" ftr)
  (setq krok (/ (- X0H X0D) kroku))
  (setq x 0)
  (load "z29") ; inicializace grafu
  (load "z30") ; nacteni procedury graf()
  (setq citac 0)
  (while (/= citac kroku)
```

```
    (progn
      (vypmot)
      (vypcelm)
      (vypocgr)
      (kresan)
      (graf)
      (setq x (+ x krok))
      (setq citac (+ 1 citac))
    )
  )
```

```
) ; konec while
```

```
; nastaveni tloustky a typu interpolace grafu
```

```
(setq citac 1)
(while (/= nil (nth citac jegraf))
  (if (= 1 (nth citac jegraf))
    (command "pedit" (eval (nth citac minent)) interp "w" tlgraf "")
  )
  (setq citac (+ 1 citac))
)
```

```
(setq citac (+ 1 citac))
) : konec while
```

```
(load "z28") : kontrola cepu, motoru a uchop. sily
```

```
(command "layer" "s" puvrstva "")
) : konec GRAFMOT()
```

8

```
(defun VYPOCGR() : vypocet hodnot pro grafy
```

```
(setq
```

```
FU (abs (/ (* (* L4 (/ fmz (* 2 (cos alfa3)))) (sin (- alfa3 alfa4))) (+ (*
```

```
(cos alfa5) L5) (* (cos alfa6) L6))))
FM (abs (* (* 2 (/ (* (+ (* (cos alfa5) L5) (* (cos alfa6) L6)) +uz) (* L4
```

```
(sin (- alfa3 alfa4)))) (cos alfa3)))
MI (+ (- pi alfa3) alfa4)
```

```
DX1 (abs (- x1lopt x11))
```

```
DX2 (abs (* ftr (if Y9 Y9 0)))
```

```
D (* 2 RO)
```

```
) : konec setq : prumer uchop. objektu
```

```
(if (= typcel 1)
```

```
(setq FI 0)
```

```
(setq FI (+ (- (/ pi 2) (/ beta3 2)) alfa6)) : kruhovy
```

```
) : prizmaticky
```

```
) : konec VYPOCGR
```

```
: nastaveni zavislosti grafu
```

```
: graf cislo: 0 1 2 3 4 5 6 7 8 9 10
```

```
(setq vztahy (list 0 'FU 'FM 'MI 'FI 'DX1 'DX2 'X 'FU 'MI 'FM))
```

```
(setq vztahx (list 0 'D 'D 'D 'D 'D 'D 'D 'X 'X 'X))
```

```
: entity krivek grafu
```

```
(setq minent (list 0 'me0 'me1 'me2 'me3 'me4 'me5 'me6 'me7 'me8 'me9))
```

```
: posledni body krivek grafu
```

```
(setq minbod (list 0 'mb0 'mb1 'mb2 'mb3 'mb4 'mb5 'mb6 'mb7 'mb8 'mb9))
```

```
: ***** samospusteni
```

```
(grafmot)
```

```
(setq grafmot nil
```

```
graf nil
```

```
vypocgr nil
```

```
vztahx nil
```

```
vztahy nil
```

```
minent nil
```

```
minbod nil
```

```
)
```

Z29.LSP

```
(defun GRAFIN(/ cit citac tedx tedy rozdy f34m f34u f5u f5m)
```

```
(setq fu (if fu fu 0)
```

```
fm (if fm fm 0)
```

```
mi (if mi mi 0)
```

```
fi (if fi fi 0)
```

```
dx1 (if dx1 dx1 0)
```

```

dx2      (if dx2 dx2 0)
)

(setq cit 1)
(while (/= nil (nth cit jegraf))      ; nastaveni noveho grafu
  (set (nth cit minbod) 0)
  (setq cit (+ 1 cit))
)
(if (null enx) (load "z25"))      ; dodelat neexistujici hlaseni
(setq D DOPT) (vypcelpr) (setq X11OPT X11)      ; vypocet X11OPT

(setq citac 0 x 0)      ; vypocty meznich hodnot
(while (/= citac kroku)
  (progn
    (vypmot)
    (vypcelm)
    (vypocqr)
    (setq f34u (abs (/ fm (* 2 (cos alfa3))))
          f34m (abs (/ fmz (* 2 (cos alfa3))))
          f5u (abs (- (sqrt (+ (expt f34u 2) (expt fuz 2))) (* (*
(* 2 f34u) fuz) (cos (+ alfa3 (/ pi 2))))))
          f5m (abs (- (sqrt (+ (expt f34m 2) (expt fu 2))) (* (*
* 2 f34m) fu) (cos (+ alfa3 (/ pi 2))))))
    )
    (if (= citac 0)      ; prvotni prednastaveni
      (progn
        (setq grminx (list 0)
              grminy (list 0)
              grmaxx (list 0)
              grmaxy (list 0)
              fmm fm
              fum fu)
        )
      (do 60 se nekresli)
    )
    (setq cit 1)
    (while (nth cit vztahy)
      (setq grminx (append grminx (list (eval (nth cit vz
tahn))))
            grminy (append grminy (list (eval (nth cit vz
tahn))))
            grmaxx (append grmaxx (list (eval (nth cit vz
tahn))))
            grmaxy (append grmaxy (list (eval (nth cit vz
tahn))))
            cit (+ 1 cit)
      )
    )
    (setq fc3 (if (> f34m f34u) f34m f34u)
          fc4 fc3
          fc5 (if (> f5m f5u) f5m f5u)
    )
    ; konec prvot. prednast.

    (progn      ; aktualizace
      (setq fc3 (if (> f34m fc3) f34m fc3) ; max. sily v c
            fc3 (if (> f34u fc3) f34u fc3)
            fc4 fc3
            fc5 (if (> f5m fc5) f5m fc5)
            fc5 (if (> f5u fc5) f5u fc5)
            fmm (if (> fmm fm) fmm fm)
            fum (if (< fum fu) fum fu)
      )
      (setq cit 1)
      (while (nth cit vztahy)
        (setq tedx (eval (nth cit vztahn))
              tedy (eval (nth cit vztahy))
        )
        (set (nth cit minbod) 0)      ; zacatek nove polyli
        (if (< tedx (nth cit grminx)) (setq grminx (nastav
cit grminx tedx)))
        (if (< tedy (nth cit grminy)) (setq grminy (nastav
cit grminy tedy)))
      )
    )
  )
)

```



```

(mapcar (lambda(a1 a2 / ed) ; aktualizace hlaseni
  (setq ed (entget (eval a1)))
  (setq ed (subst (cons 1 (rtos (eval a2))) (assoc 1 ed) ed))
  (entmod ed)
)
(enx end enfu enfm enmi enfi endx1 endx2)
(x (* 2 r0) fu fm mi fi dx1 dx2)
)

```

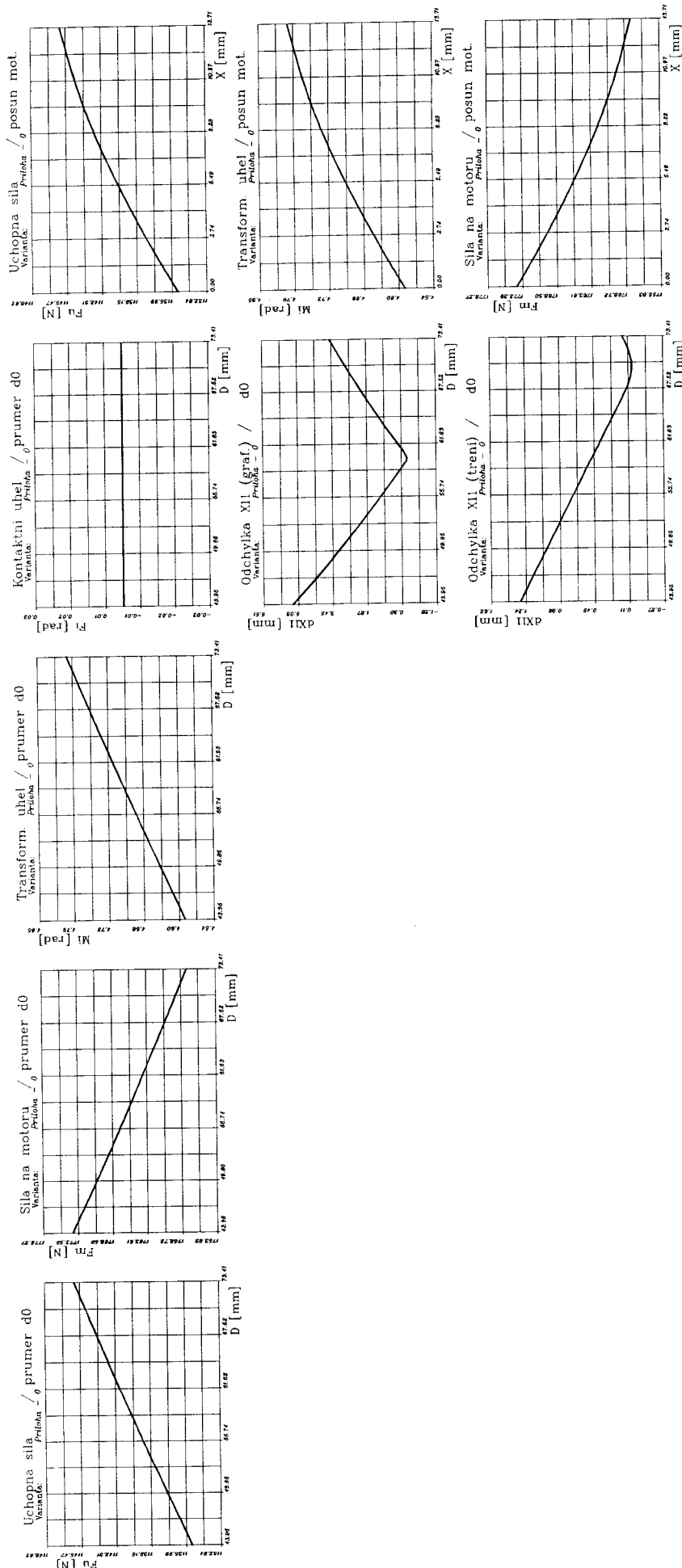
11

```

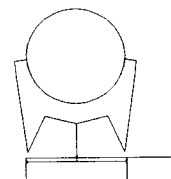
(setq citac 1) ; grafy
(while (/= nil (nth citac jefraf))
  (if (= 1 (nth citac jefraf))
    (progn ; vypocet noveho bodu
      (setq tedbod (list (+ (* (- (eval (nth citac vztahx)) (nth citac g
rminx)) (/ 250 (- (nth citac grmaxx) (nth citac grminx)))) (nth citac gxd))
        (+ (* (- (eval (nth citac vztahy)) (nth citac grminy)) (/
160 (- (nth citac grmaxy) (nth citac grminy)))) (nth citac gyd))
      )
      (if (= (eval (nth citac minbod)) 0)
        (progn
          (progn
            (command "pline" tedbod (eval (nth citac minbod)) "")
            (setq tedent (entlast))
            (if (eval (nth citac minent))
              (command "pedit" tedent "w" tlgraf "j" (eval (nth citac minent)
) tedent "" ""))
          )
        )
        ; konec if
        (if (/= (eval (nth citac minbod)) 0) (set (nth citac minent) (entlast))
)
        (set (nth citac minbod) tedbod)
        ; konec progn
      ) ; konec if
      (setq citac (+ 1 citac))
    ) ; konec while
  ) ; konec graf()
)

```

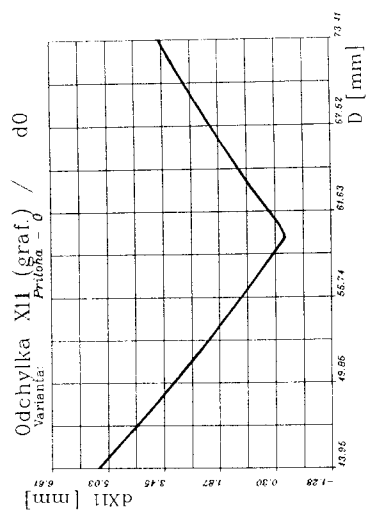
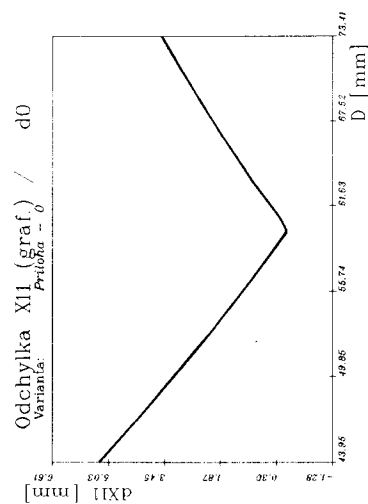
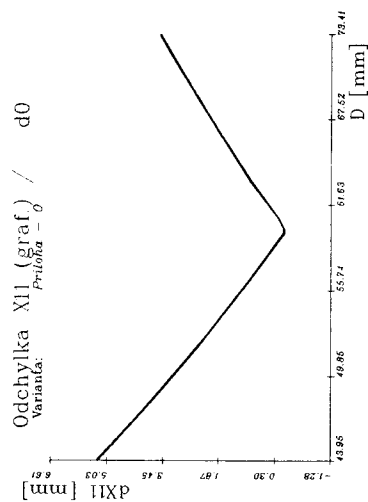
Příloha č. 4



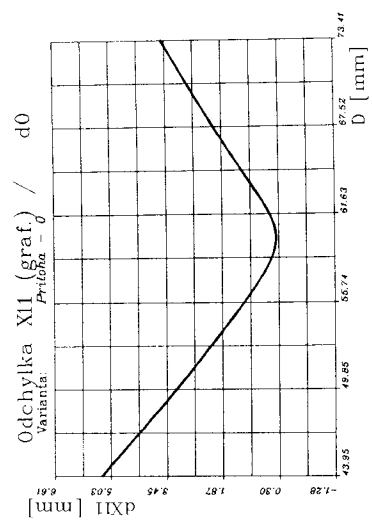
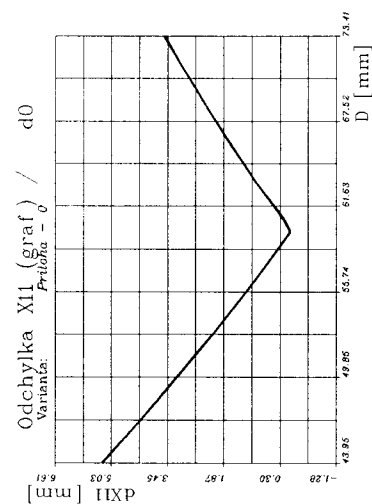
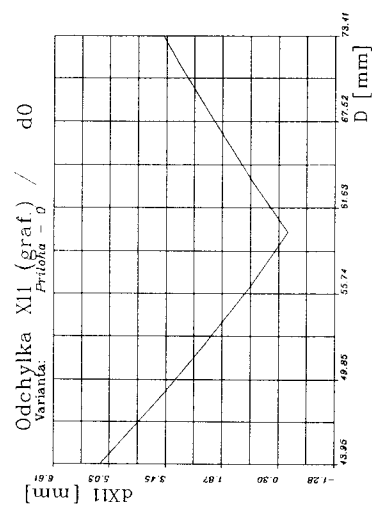
Posun motoru X [mm] = 13.7149
 Uchopná síla F_u [N] = 73.4148
 Uchopná síla F_m [N] = 173.5892
 Síla na motoru F_m [N] = 173.5892
 Síla na motoru M_t [rad] = 4.6008
 Kontaktní úhel F_t [rad] = 0.0000
 Úchylka struch (graf.) [mm] = 3.5446
 Úchylka struch (trent) [mm] = 0.1707



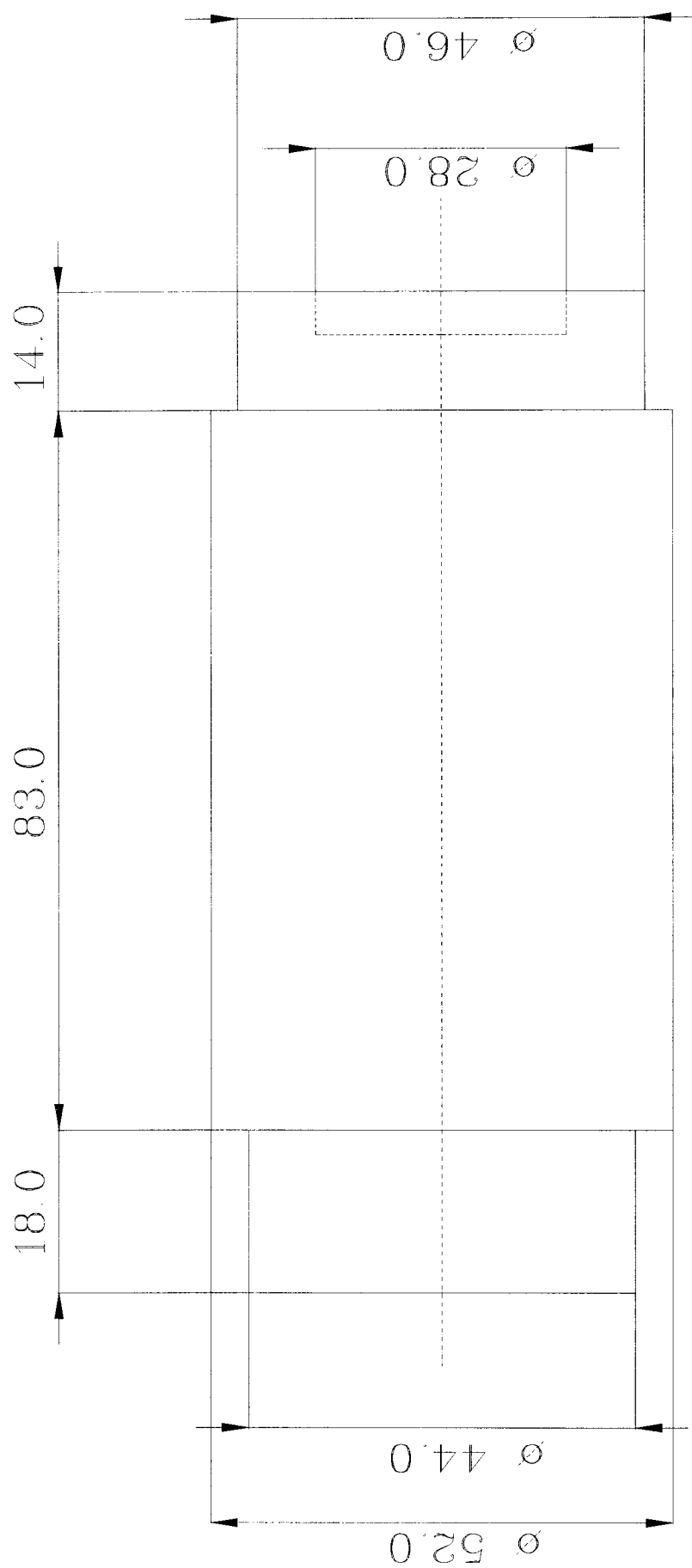
Druhý graf:



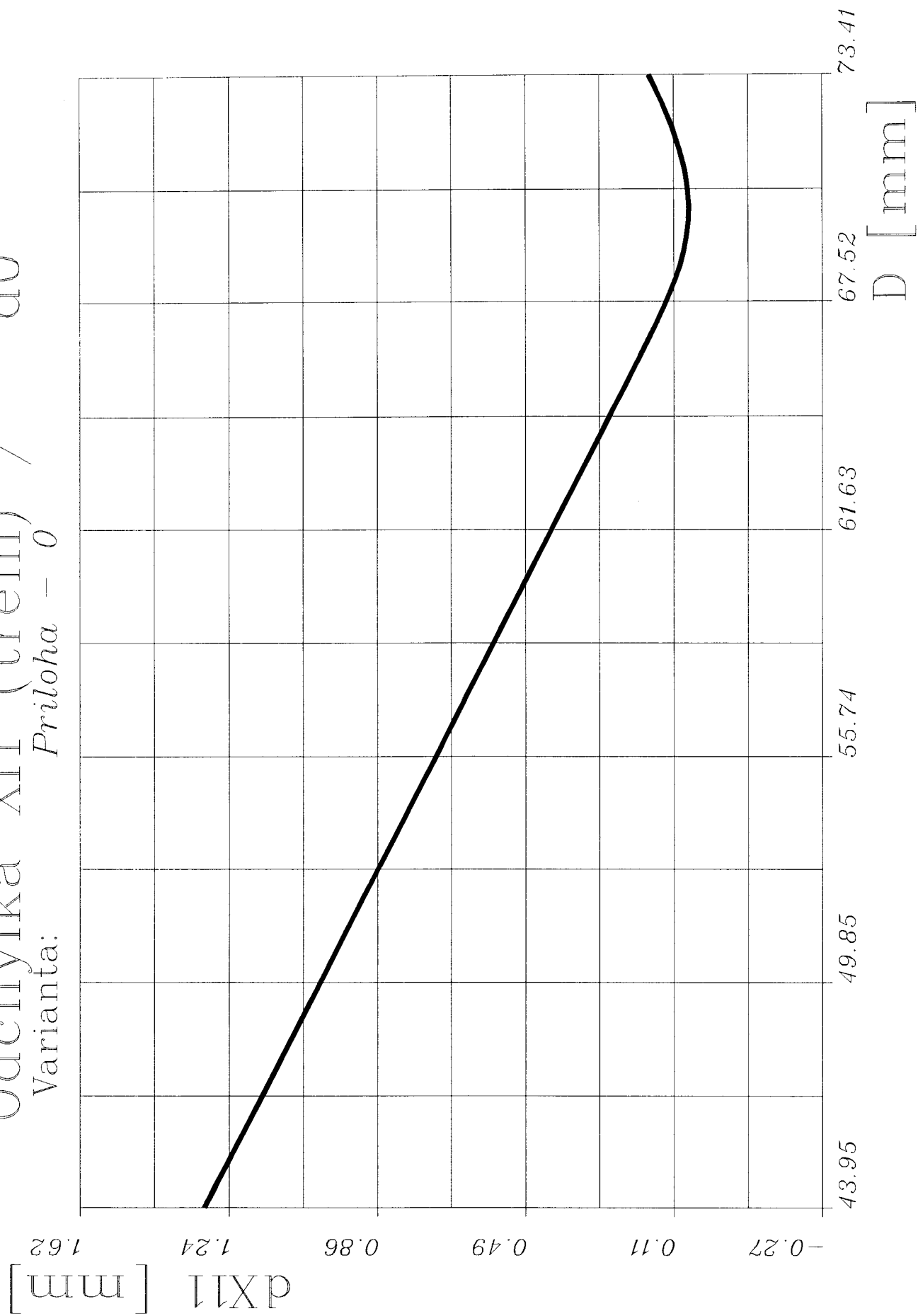
Druhý interpolační graf:



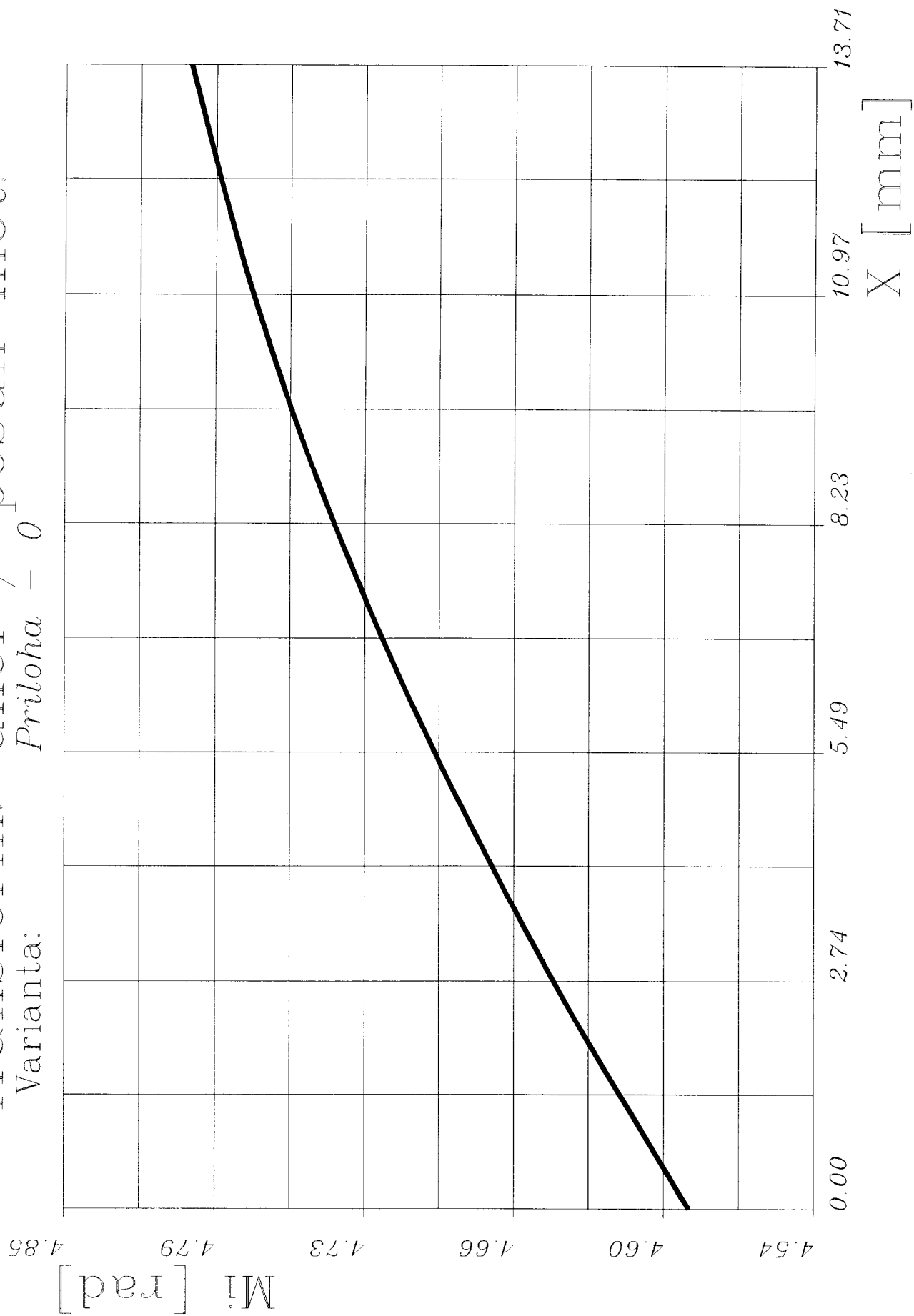
Manipulovaný objekt



Odchylka X11 (treni) / d0
 Varianta: Priloha - 0



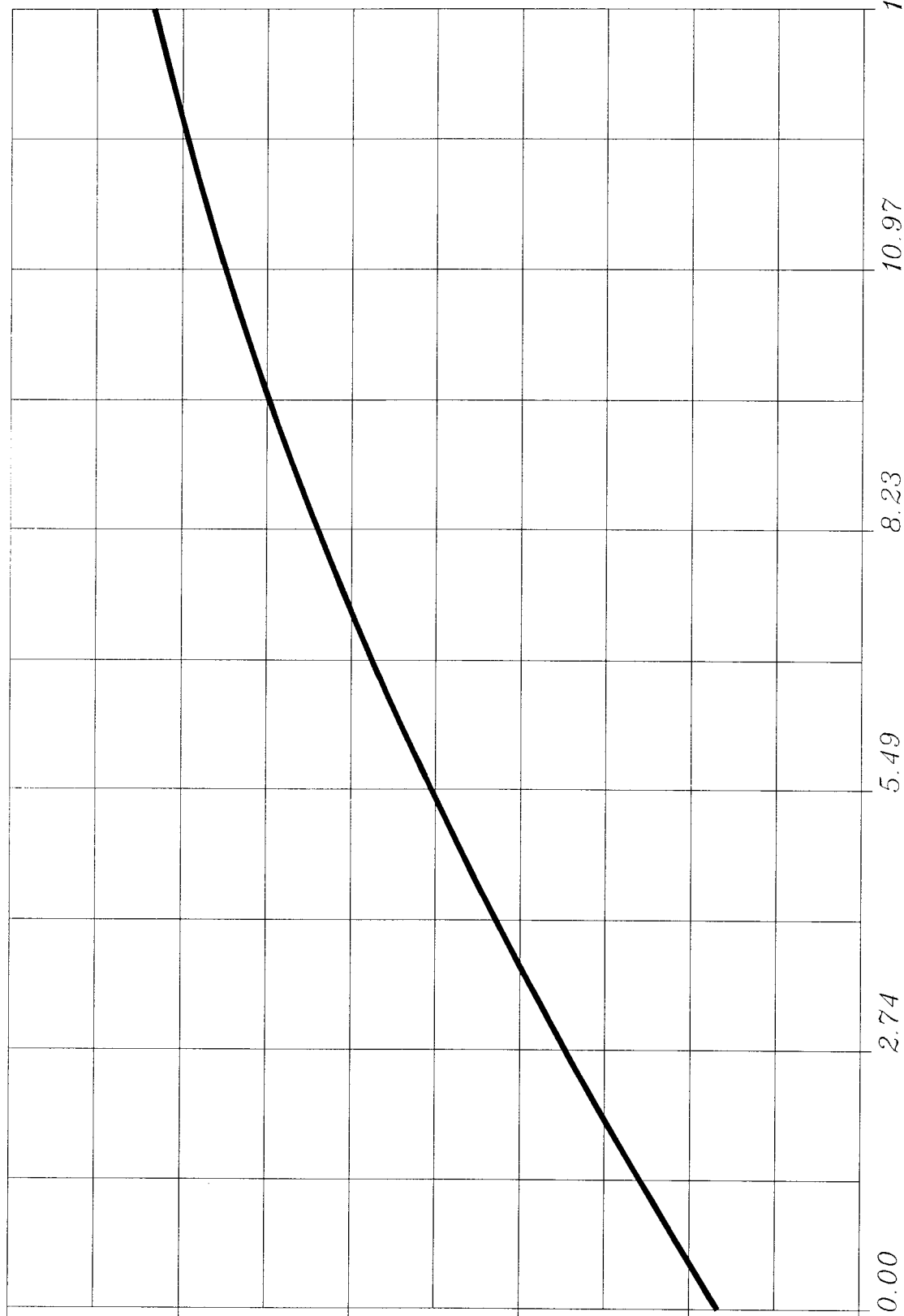
Transform. uhel / posun mot.
 Priloha - 0



Uchopna sila / posun mot.
 Varianta: Priloha - 0

1148.62

Fu [N]



X [mm]

Posuv motoru / průměr d0
 Varianta: Příloha -- 0

17.14

X [mm]

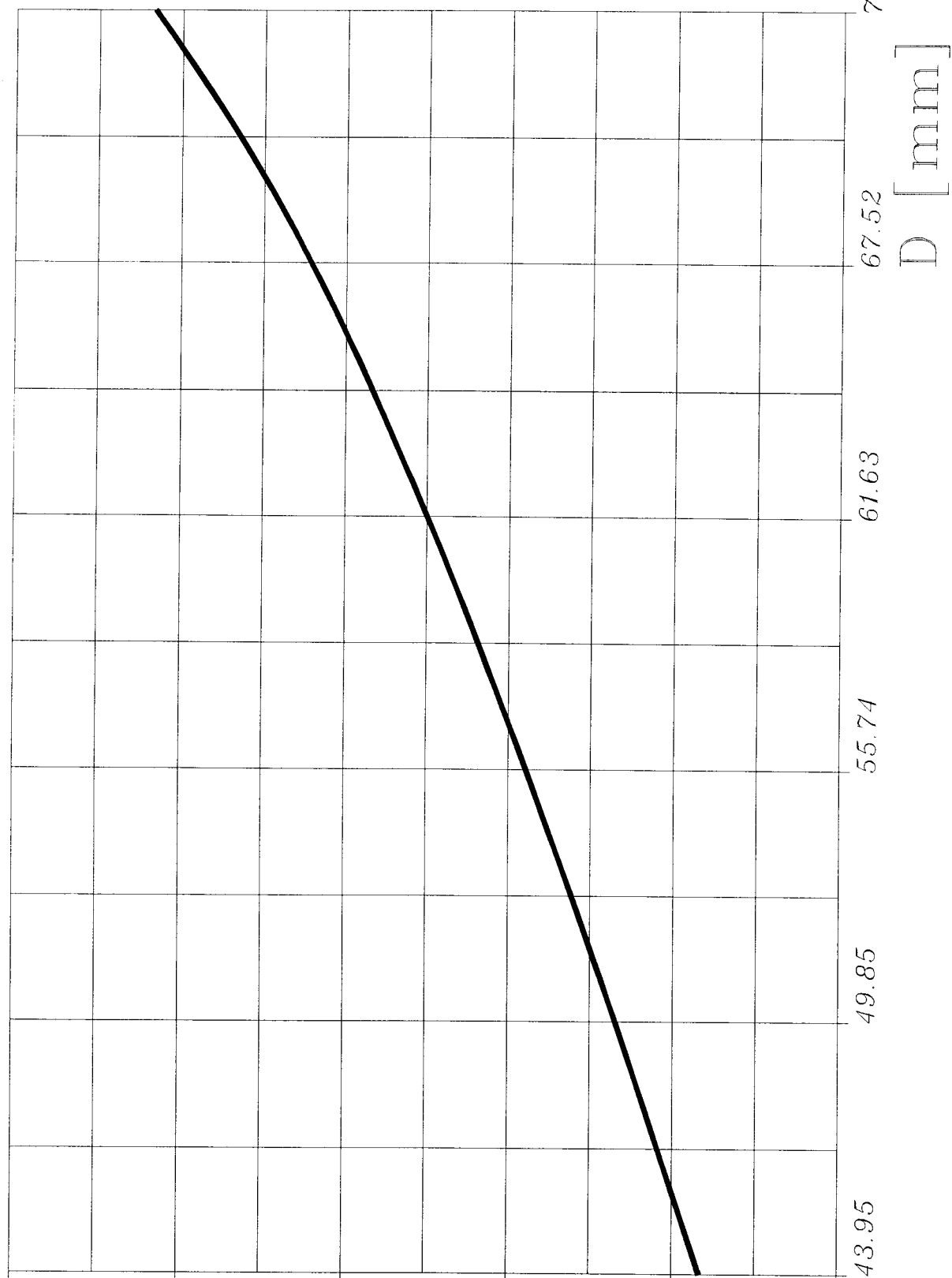
13.03

8.91

4.80

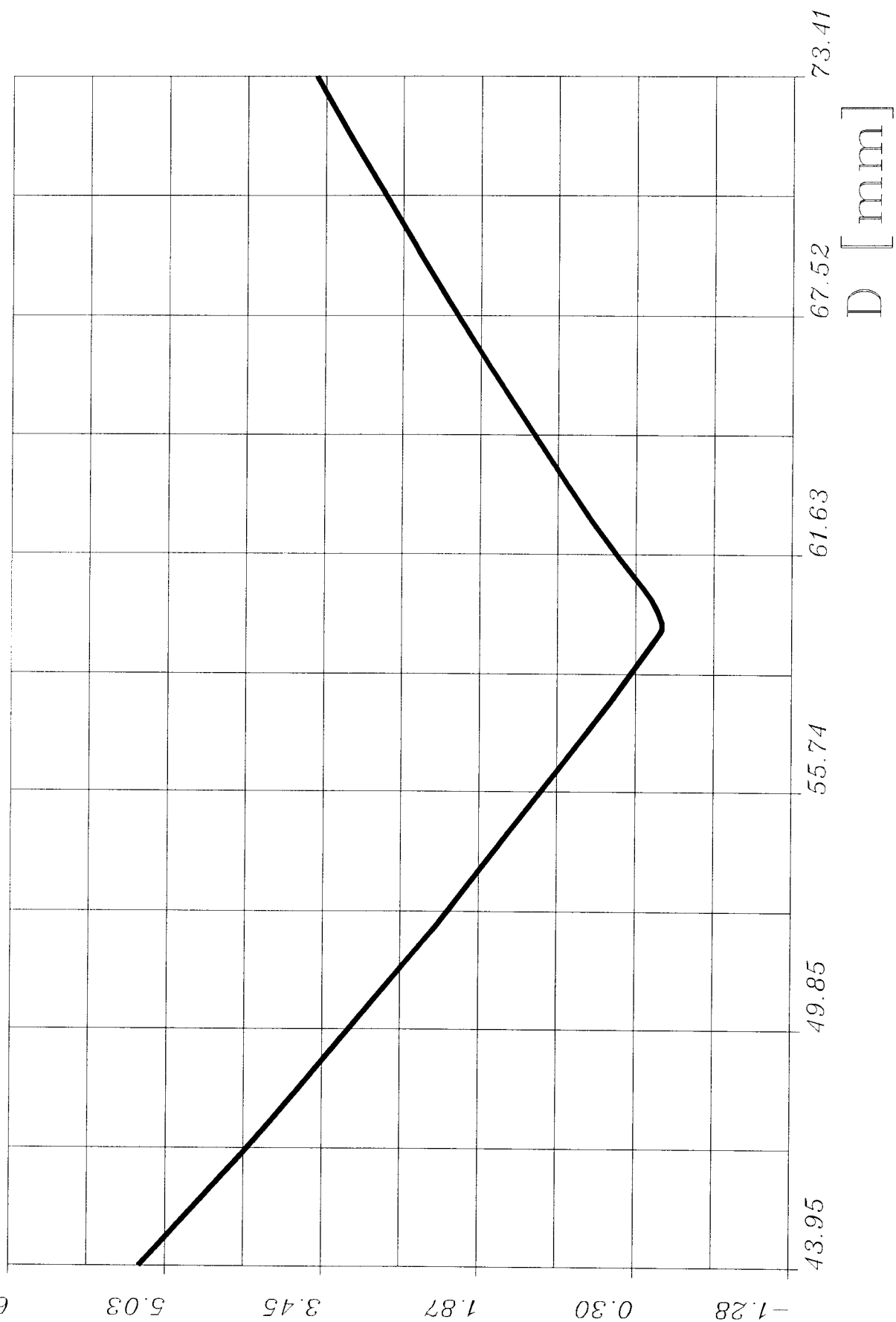
0.69

-3.43



Odchylka X11 (graf.) / d0
 Varianta: Příloha - 0

dX11 [mm]



Kontaktní uhel / průměr d0
 Varianta: Příloha – 0

0.03

F_i [rad]

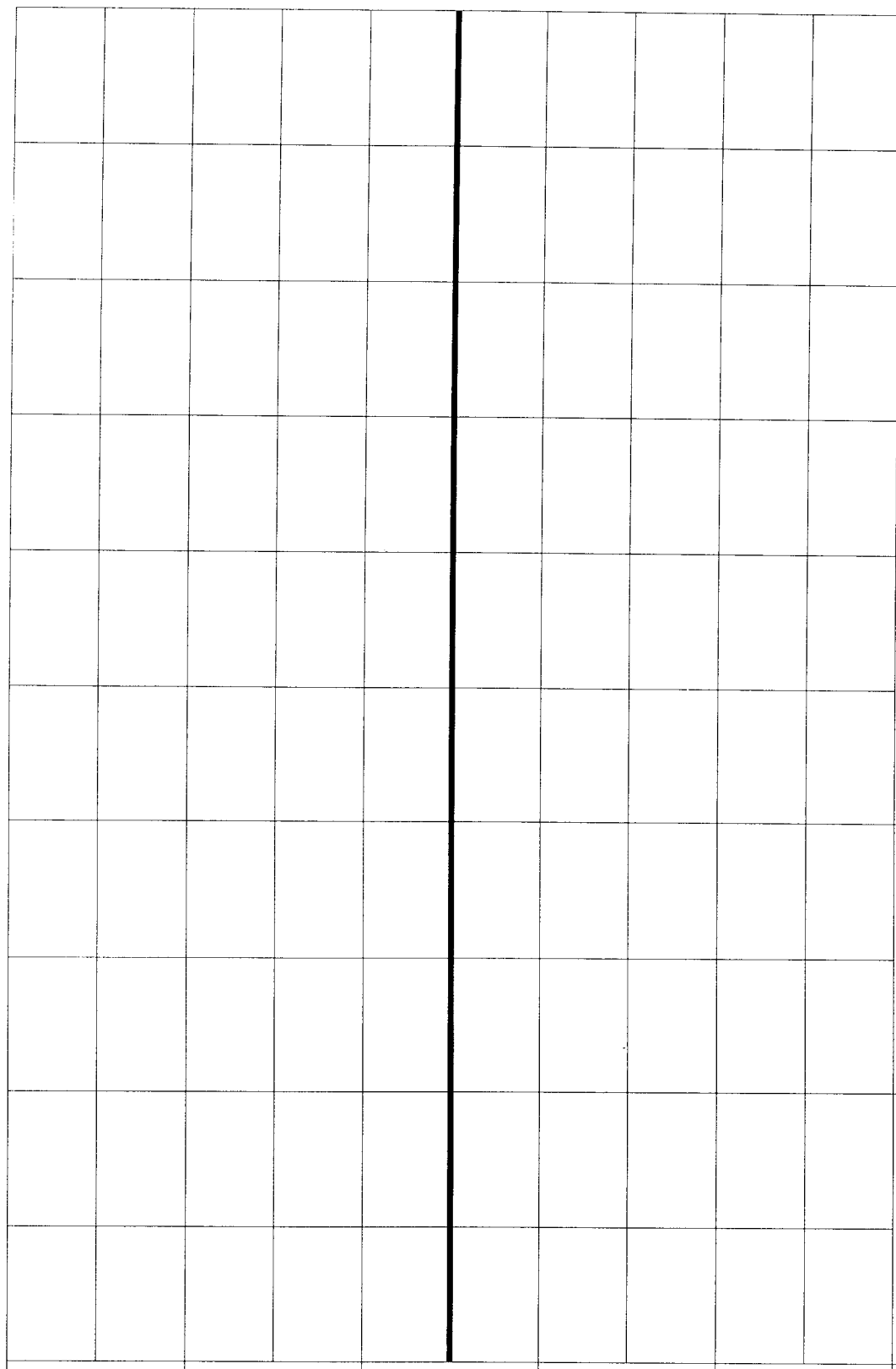
0.02

0.01

-0.01

-0.02

-0.03



43.95

49.85

55.74

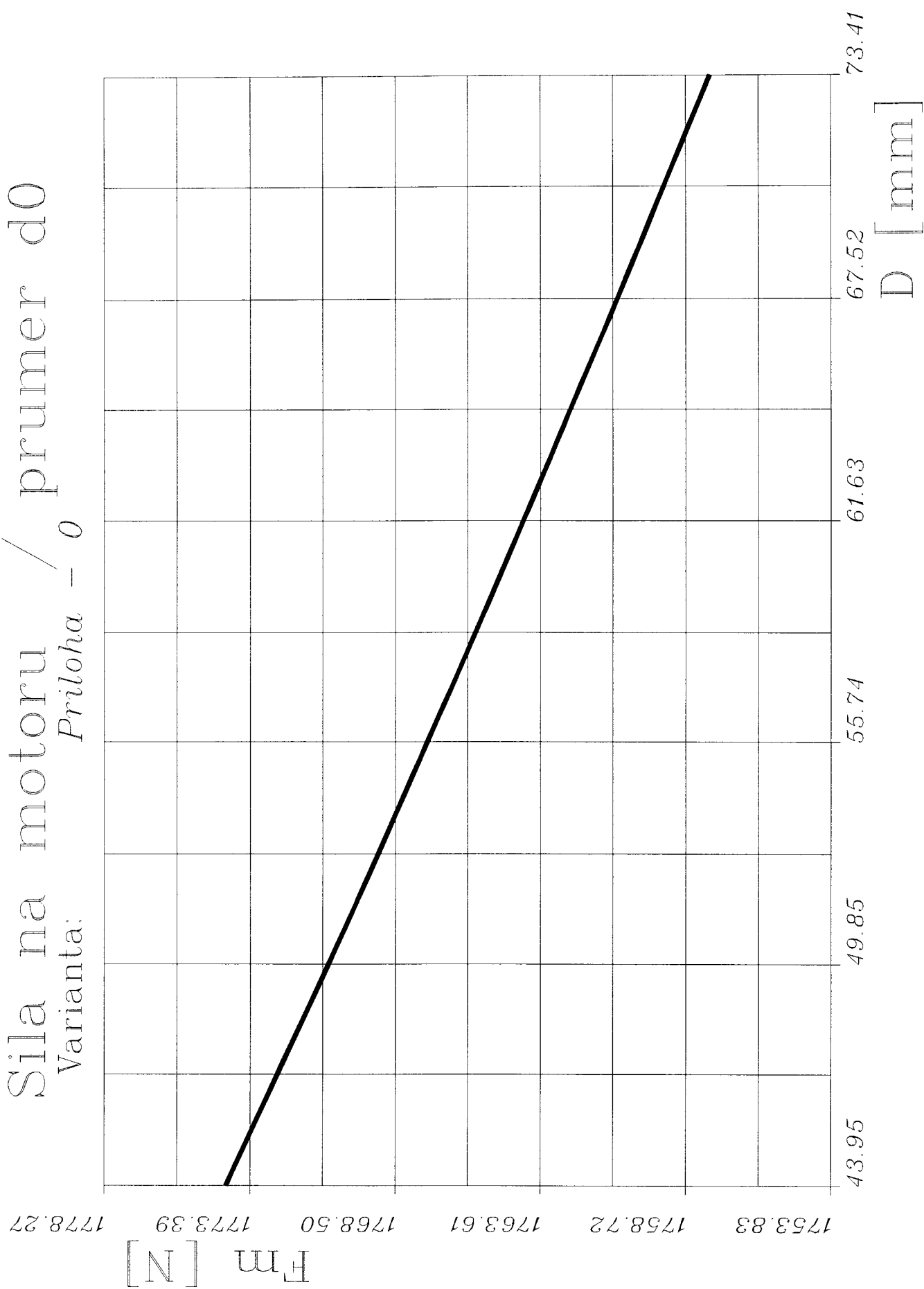
61.63

67.52

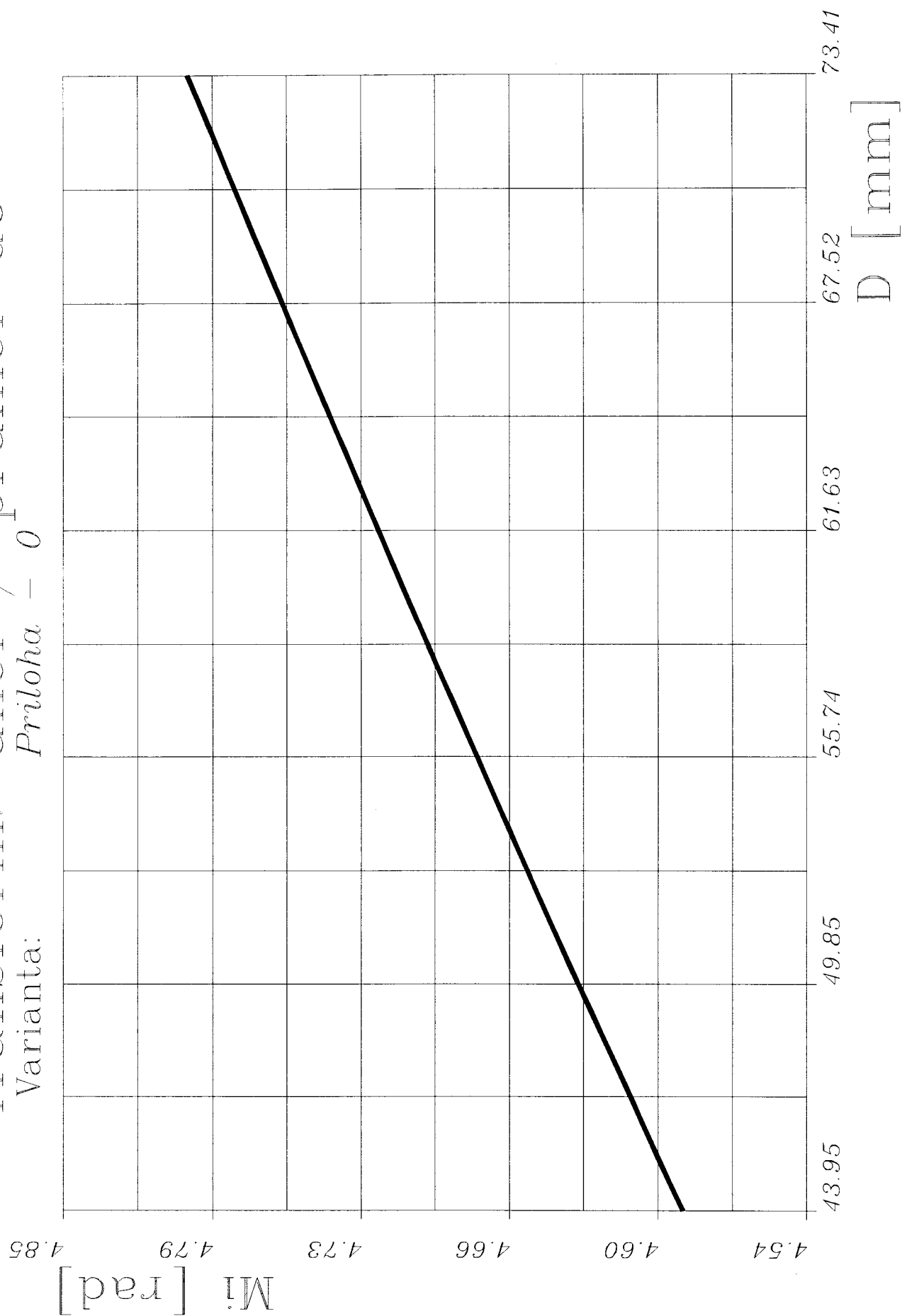
73.41

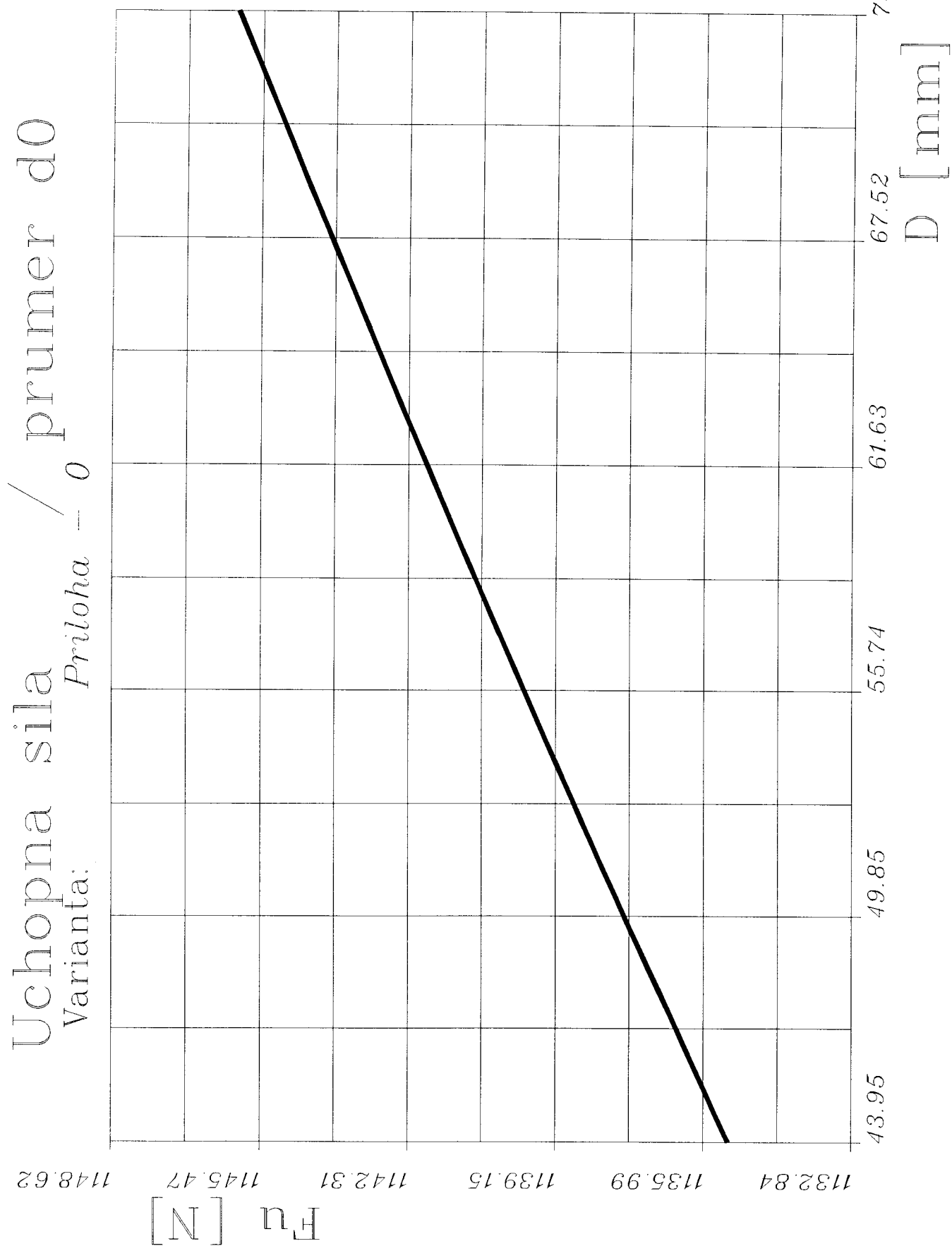
D [mm]

Síla na motoru / průměr d0
 Varianta: Příloha – 0

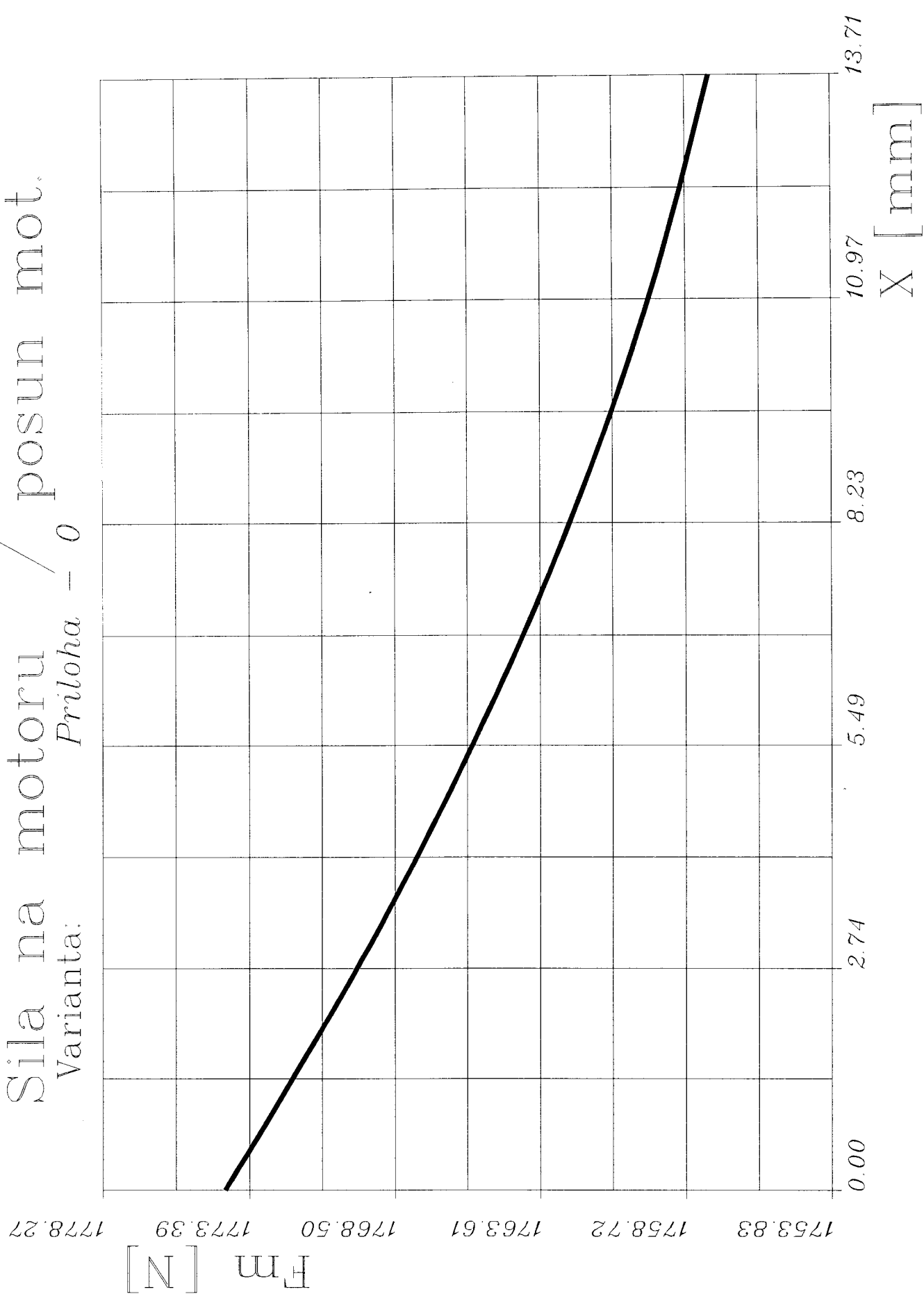


Transform. uhel / průměr d0
 Varianta: Priloha - 0

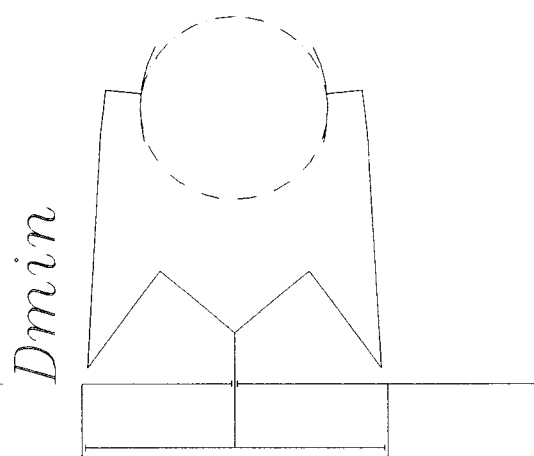
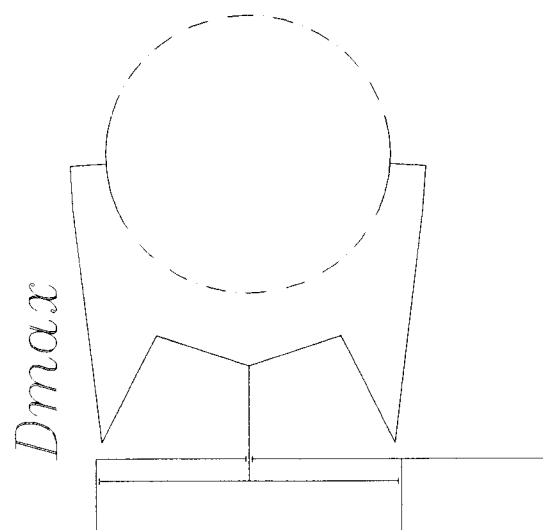
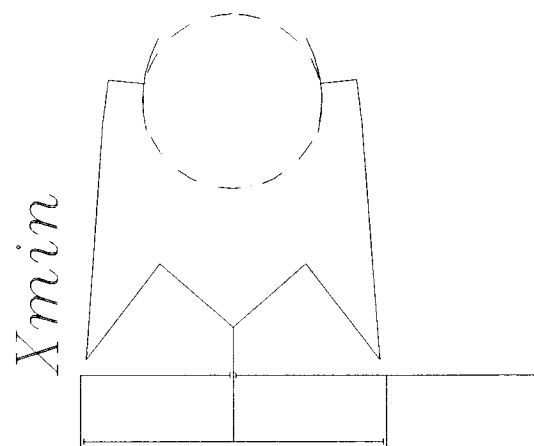
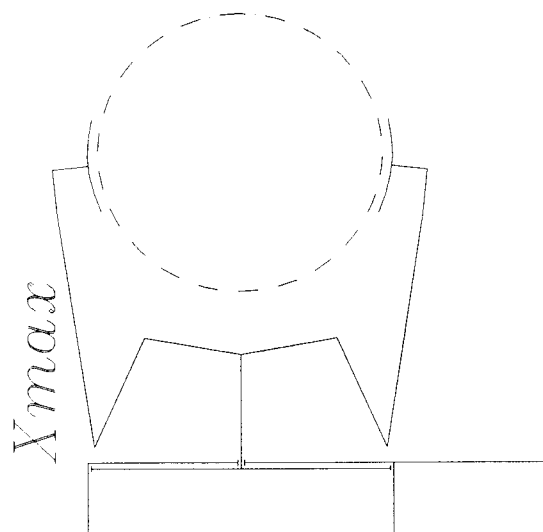
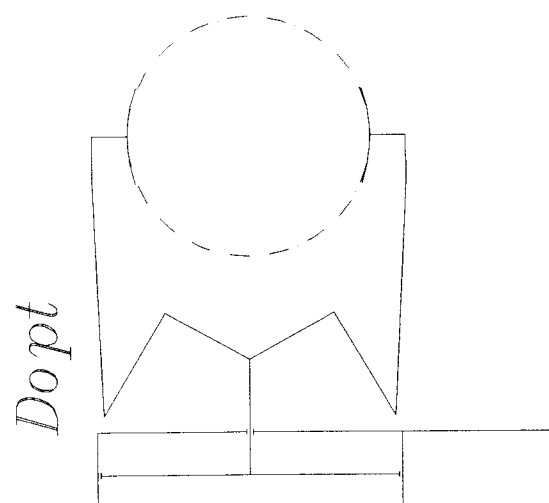




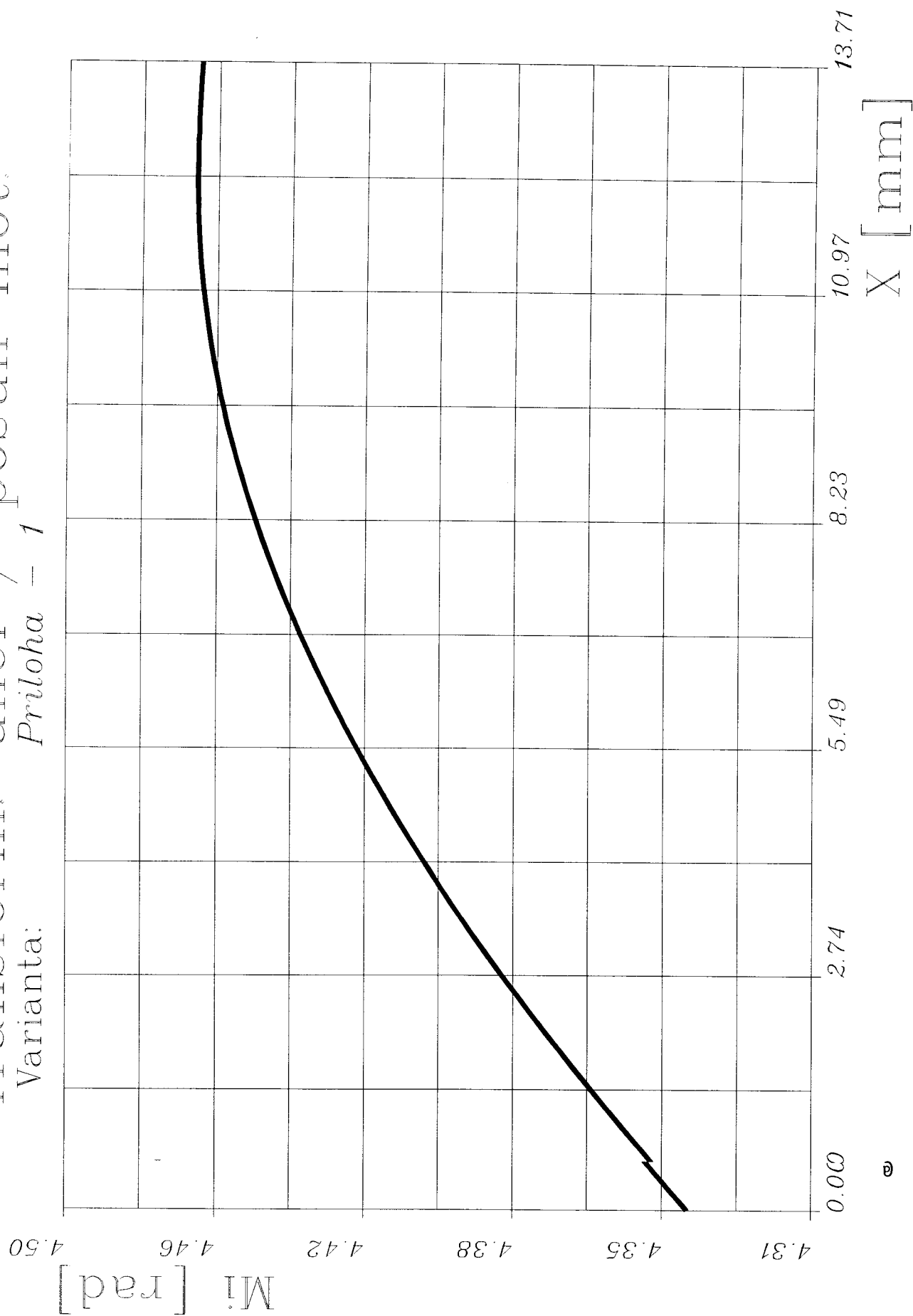
Sila na motoru / posun mot.
 Varianta: Příloha - 0



Příloha č. 10

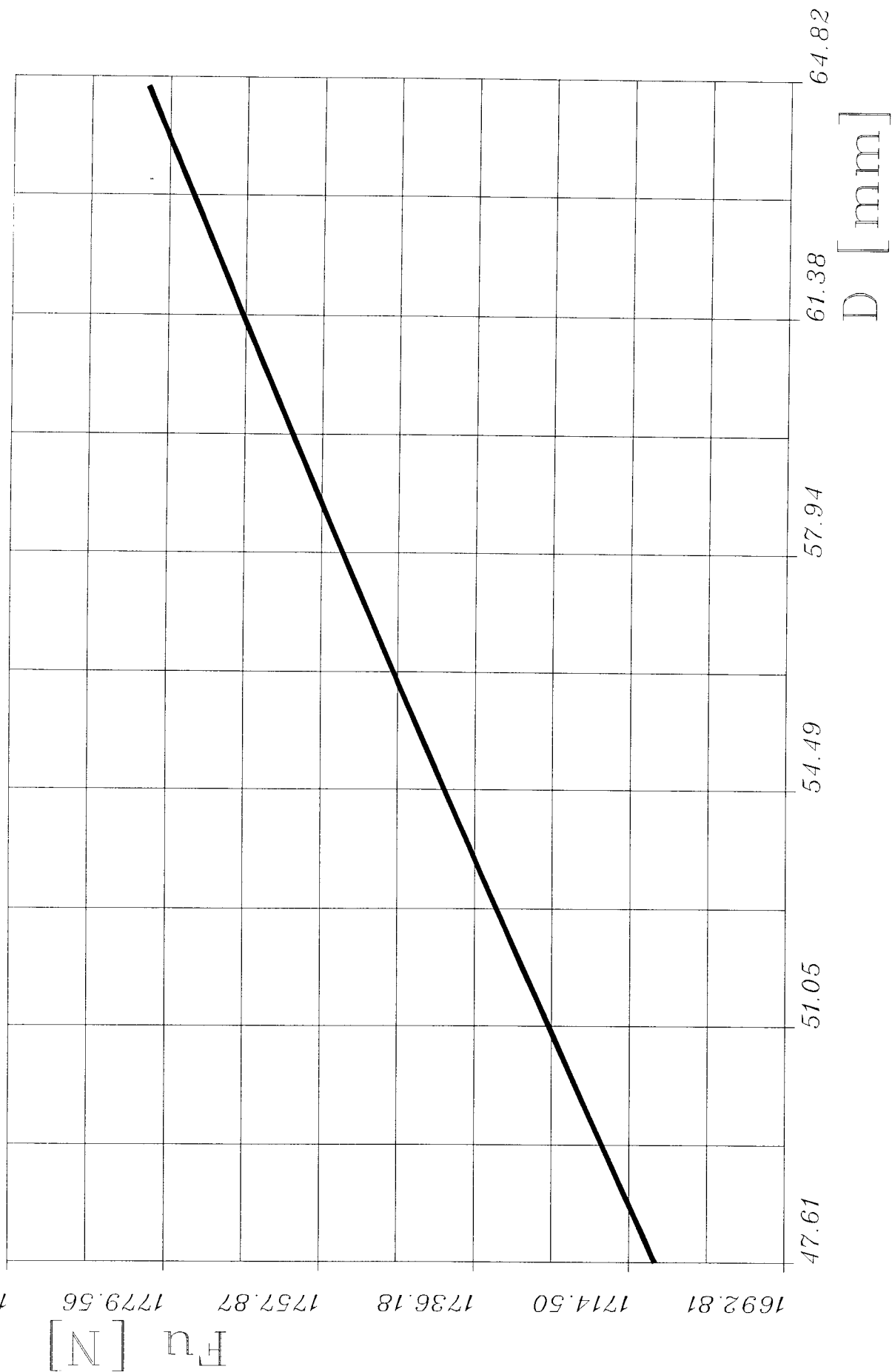


Transform. uhel / posun mot.
 Varianta: *Priloha - 1*

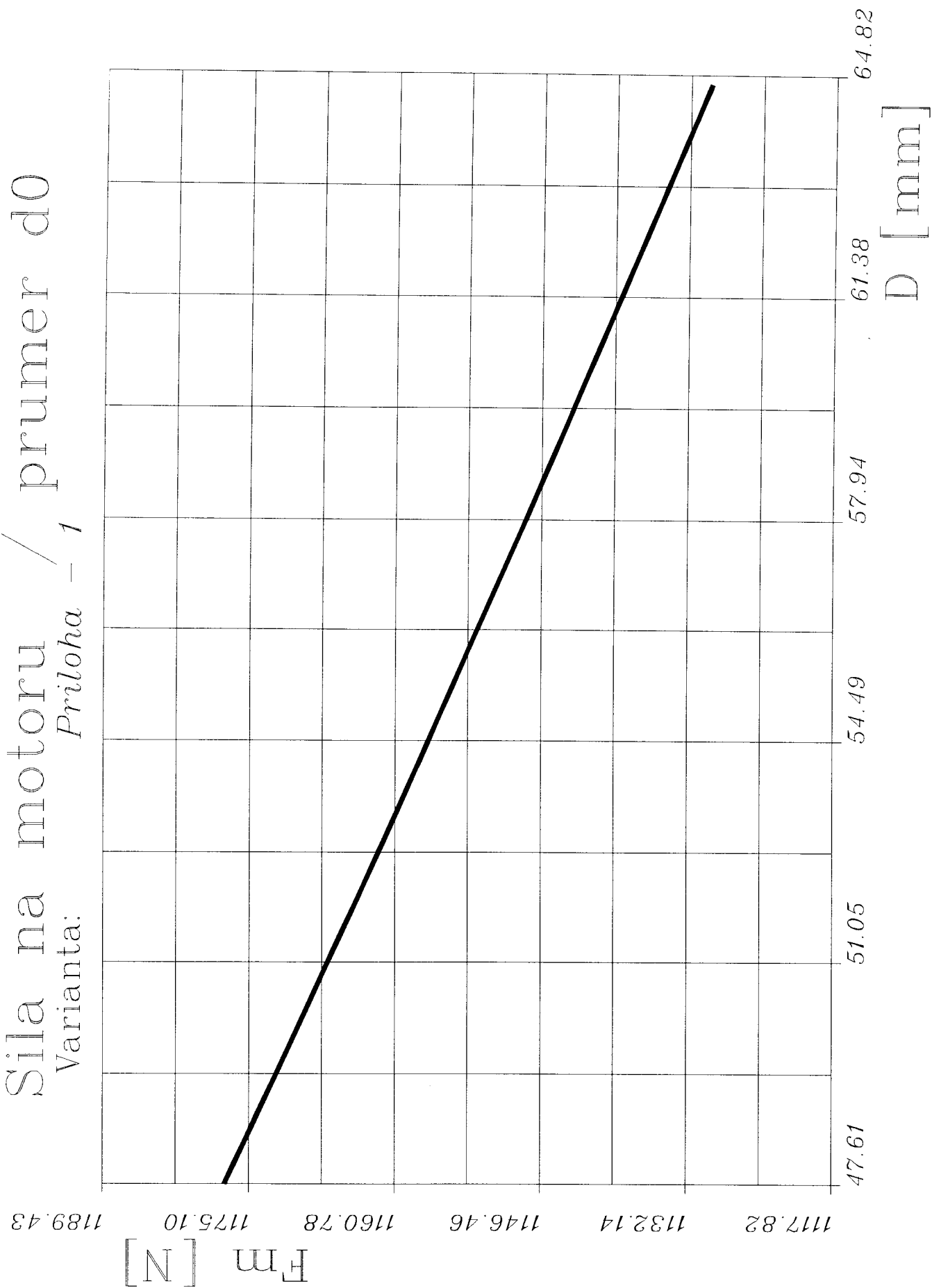


Uchopna sila / prumer d0
Varianta: Priloha - 1

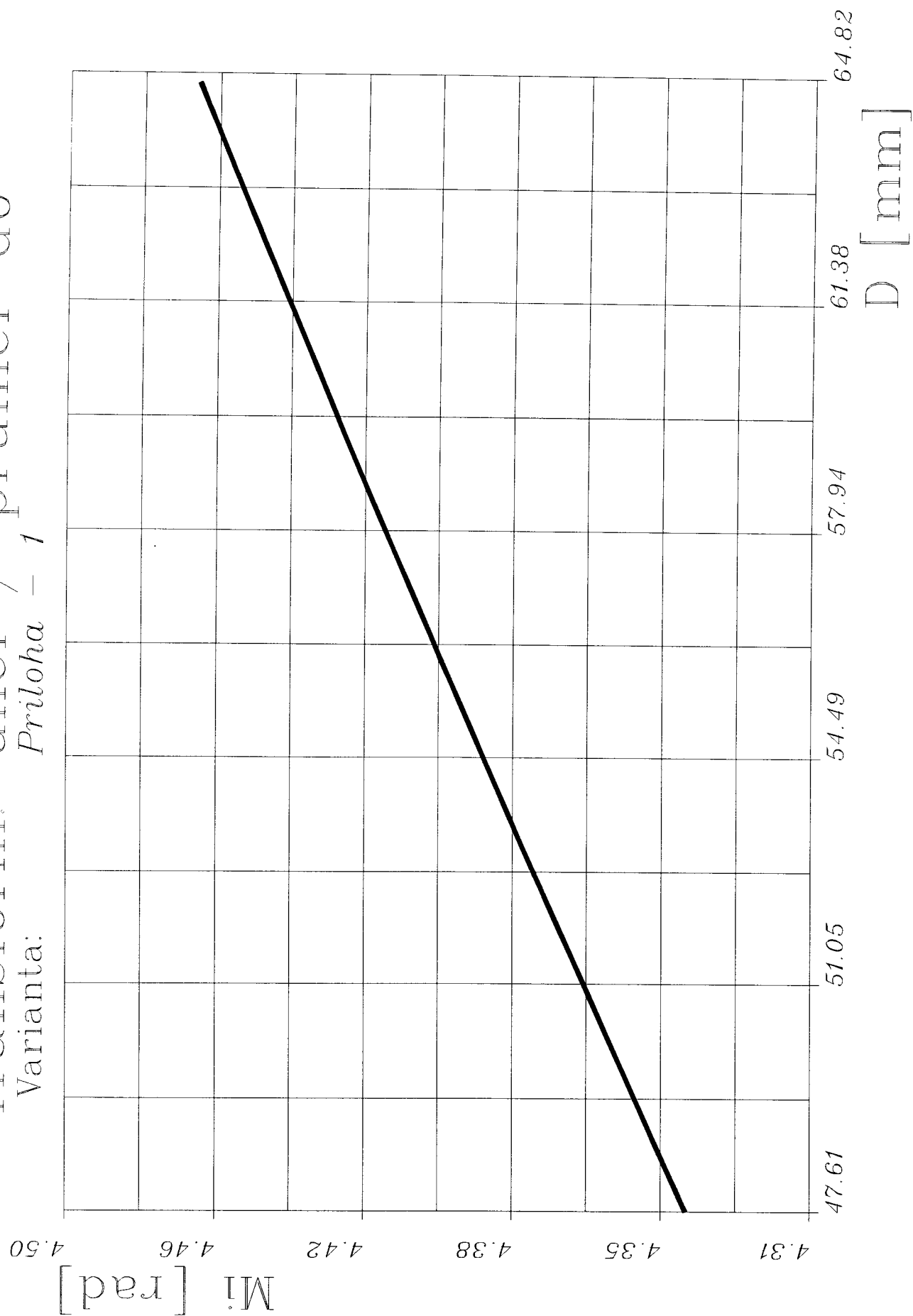
F_u [N]



Síla na motoru / průměr d0
Varianta: Příloha - 1

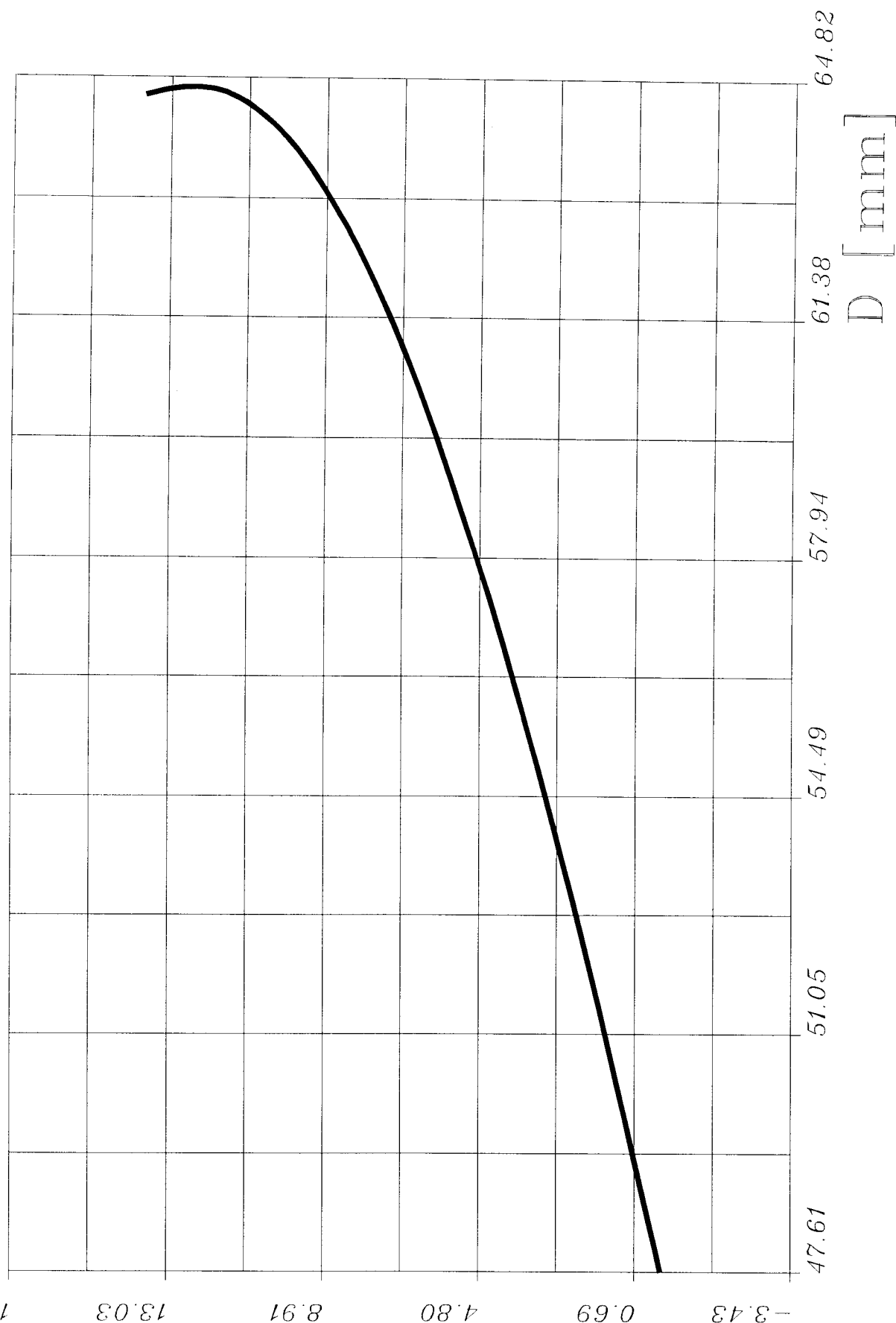


Transform. uhel / průměr d0
Varianta: Příloha - 1



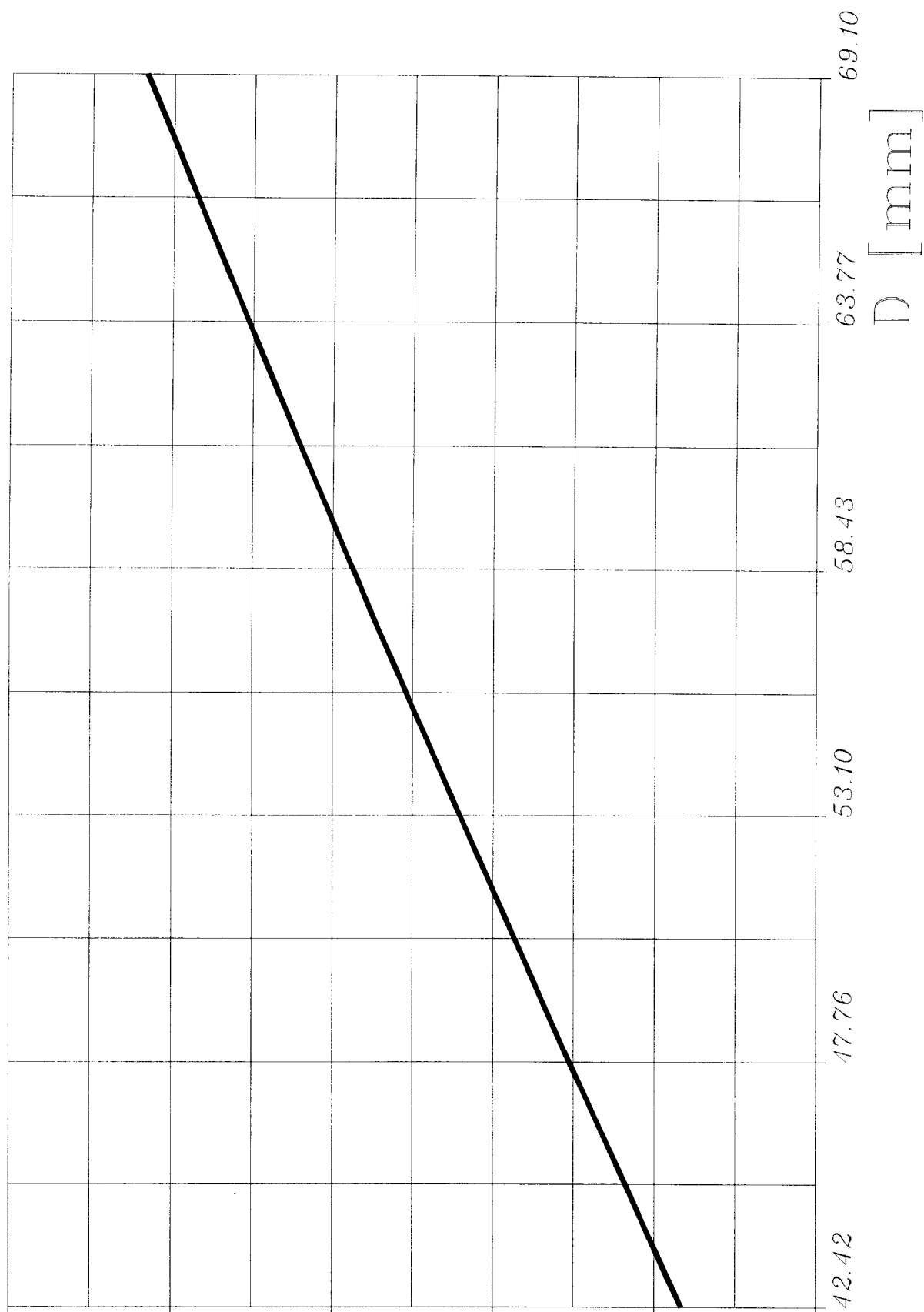
Posuv motoru / průměr d0
Varianta: Příloha - 1

X [mm]



Kontaktní uhel / průměr d0
 Varianta: Příloha - 4

F_i [rad]



Odchylka X11 (treni) / d0
 Varianta: Příloha - 4

dX11 [mm]

0.03

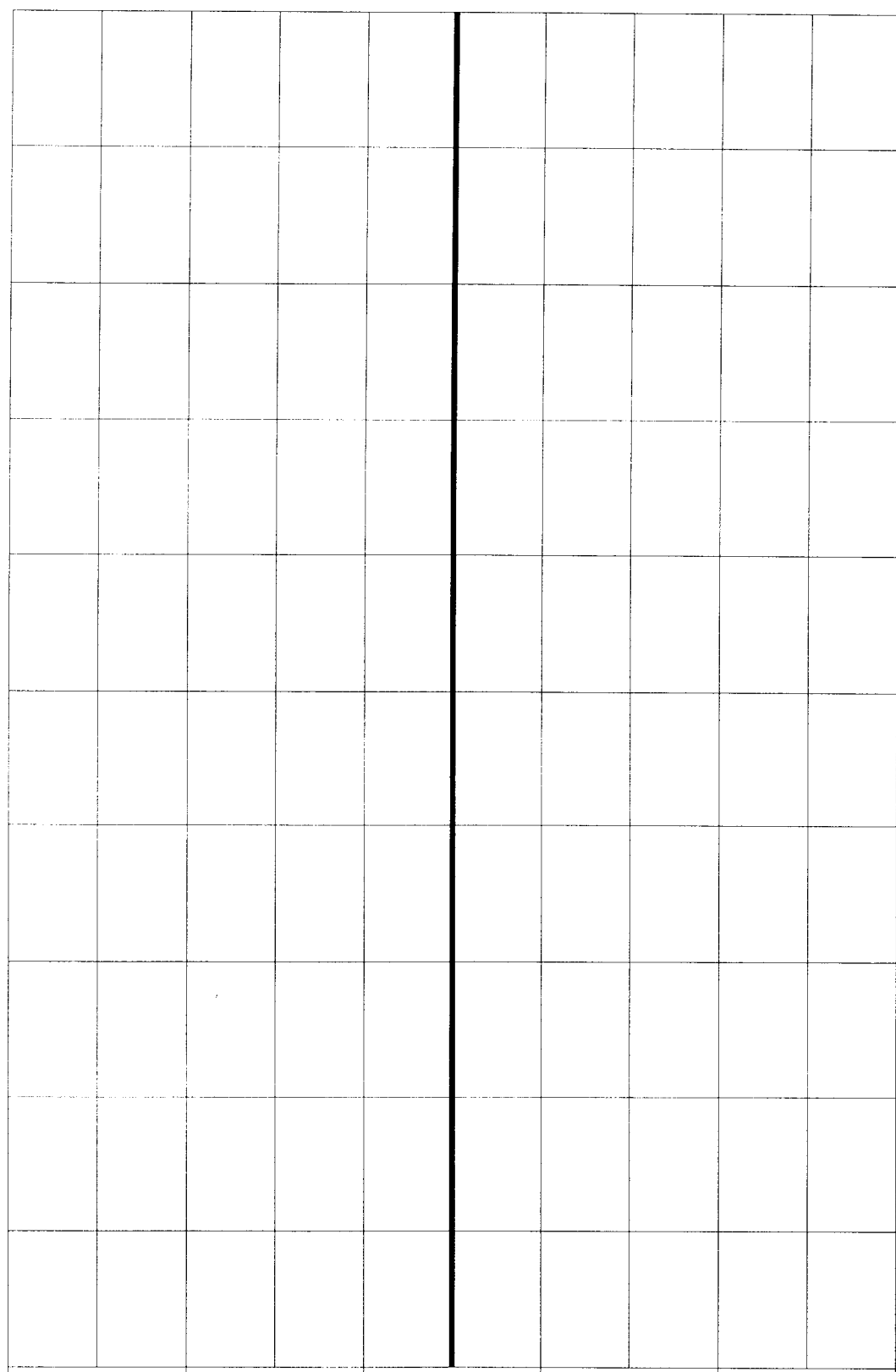
0.02

0.01

-0.01

-0.02

-0.03



42.42

47.76

53.10

58.43

63.77

69.10

D [mm]

Odchylka X11 (graf.) / d0
 Varianta: Příloha - 4

dX11 [mm]

