1

OF/DE

1

PM-1 3 1/2" x 4" PHOTOGRAPHIC MICROCOPY TARGET
NBS 1010a ANSI/ISO #2 EQUIVALENT

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Number / Numéro de la notice

Call / Appel / Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

**Application of the Medial Axis Transform
in a Design Environment for Die Casting**

by

John Michael New, B.Sc.

A thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfilment of
the requirements for the degree of
Master of Engineering

**Department of Mechanical and Aerospace Engineering
Ottawa-Carleton Institute
for Mechanical and Aerospace Engineering**

Carleton University
Ottawa, Ontario
April 19, 1993

© copyright
1993, J. Michael New

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Votre titre / Votre référence

Chapitre / Autres références

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-84137-0

Canada

Name

J. Michael New

Dissertation Abstracts International is arranged by broad, general subject categories. Please select the one subject which most nearly describes the content of your dissertation. Enter the corresponding four-digit code in the spaces provided.

Mechanical Engineering

SUBJECT TERM

0548

SUBJECT CODE

U·M·I

Subject Categories

THE HUMANITIES AND SOCIAL SCIENCES

COMMUNICATIONS AND THE ARTS

Architecture 0729
Art History 0377
Cinema 0900
Dance 0378
Fine Arts 0357
Information Science 0723
Journalism 0391
Library Science 0399
Mass Communications 0708
Music 0413
Speech Communication 0459
Theater 0465

EDUCATION

General 0515
Administration 0514
Adult and Continuing 0516
Agricultural 0517
Art 0273
Bilingual and Multicultural 0282
Business 0688
Community College 0275
Curriculum and Instruction 0727
Early Childhood 0518
Elementary 0574
Finance 0277
Guidance and Counseling 0519
Health 0680
Higher 0745
History of 0520
Home Economics 0278
Industrial 0521
Language and Literature 0279
Mathematics 0280
Music 0522
Philosophy of 0998
Physical 0523

Psychology 0525
Reading 0535
Religious 0527
Sciences 0714
Secondary 0533
Social Sciences 0534
Sociology of 0340
Special 0529
Teacher Training 0530
Technology 0710
Tests and Measurements 0288
Vocational 0747

LANGUAGE, LITERATURE AND LINGUISTICS

Language 0679
General 0289
Ancient 0290
Linguistics 0291
Modern 0291
Literature 0401
General 0294
Classical 0295
Comparative 0297
Medieval 0298
Modern 0316
African 0591
American 0305
Asian 0352
Canadian (English) 0355
Canadian (French) 0593
English 0311
Germanic 0312
Latin American 0315
Middle Eastern 0313
Romance 0314
Slavic and East European 0314

PHILOSOPHY, RELIGION AND THEOLOGY

Philosophy 0422
Religion 0318
General 0321
Biblical Studies 0319
Clergy 0320
History of 0322
Philosophy of 0469
Theology 0469

SOCIAL SCIENCES

American Studies 0323
Anthropology 0324
Archaeology 0326
Cultural 0327
Physical 0310
Business Administration 0272
General 0770
Accounting 0454
Banking 0338
Management 0385
Canadian Studies 0501
Economics 0503
General 0505
Agricultural 0508
Commerce Business 0509
Finance 0510
History 0511
Labor 0358
Theory 0366
Folklore 0351
Geography 0578
Gerontology 0351
History 0578
General 0578

Ancient 0579
Medieval 0581
Modern 0582
Black 0328
African 0331
Asia, Australia and Oceania 0332
Canadian 0334
European 0335
Latin American 0336
Middle Eastern 0333
United States 0337
History of Science 0585
Law 0398
Political Science 0615
General 0616
International Law and Relations 0617
Public Administration 0814
Recreation 0452
Social Work 0626
Sociology 0627
General 0938
Criminology and Penology 0631
Demography 0628
Ethnic and Racial Studies 0629
Individual and Family Studies 0630
Industrial and Labor Relations 0700
Public and Social Welfare 0344
Social Structure and Development 0709
Theory and Methods 0999
Transportation 0453
Urban and Regional Planning 0453
Women's Studies 0453

THE SCIENCES AND ENGINEERING

BIOLOGICAL SCIENCES

Agriculture 0473
General 0285
Agronomy 0475
Animal Culture and Nutrition 0476
Animal Pathology 0359
Food Science and Technology 0478
Forestry and Wildlife 0479
Plant Culture 0480
Plant Pathology 0817
Plant Physiology 0777
Range Management 0746
Wood Technology 0306
Biology 0287
General 0308
Anatomy 0309
Biostatistics 0310
Botany 0311
Cell 0312
Ecology 0313
Entomology 0314
Genetics 0315
Limnology 0316
Microbiology 0317
Molecular 0318
Neuroscience 0319
Oceanography 0320
Physiology 0321
Radiation 0322
Veterinary Science 0323
Zoology 0324
Biophysics 0325
General 0326
Medical 0327

EARTH SCIENCES

Biogeochemistry 0425
Geochemistry 0906

Geodesy 0370
Geology 0372
Geophysics 0373
Hydrology 0388
Mineralogy 0411
Paleobotany 0345
Paleoecology 0426
Paleontology 0418
Paleozoology 0985
Palynology 0427
Physical Geography 0368
Physical Oceanography 0415

HEALTH AND ENVIRONMENTAL SCIENCES

Environmental Sciences 0768
Health Sciences 0566
General 0300
Audiology 0992
Chemotherapy 0567
Dentistry 0350
Education 0769
Hospital Management 0758
Human Development 0982
Immunology 0564
Medicine and Surgery 0347
Mental Health 0569
Nursing 0570
Nutrition 0380
Obstetrics and Gynecology 0354
Occupational Health and Therapy 0381
Ophthalmology 0571
Pathology 0419
Pharmacology 0572
Pharmacy 0382
Physical Therapy 0573
Public Health 0574
Radiology 0575
Recreation 0575

Speech Pathology 0460
Toxicology 0383
Home Economics 0386

PHYSICAL SCIENCES

Pure Sciences 0485
Chemistry 0749
General 0486
Agricultural 0487
Analytical 0488
Biochemistry 0738
Inorganic 0490
Nuclear 0491
Organic 0494
Pharmaceutical 0495
Physical 0754
Polymer 0405
Radiation 0605
Mathematics 0986
Physics 0606
General 0608
Acoustics 0748
Astronomy and Astrophysics 0607
Atmospheric Science 0798
Atomic 0759
Electronic and Electricity 0609
Elementary Particles and High Energy 0610
Fluid and Plasma 0752
Molecular 0756
Nuclear 0611
Optics 0463
Radiation 0346
Solid State 0984
Statistics 0984

Applied Sciences

Applied Mechanics 0346
Computer Science 0984

Engineering 0537
General 0538
Aerospace 0539
Agricultural 0540
Automotive 0541
Biomedical 0542
Chemical 0543
Civil 0544
Electronics and Electrical 0348
Heat and Thermodynamics 0545
Hydraulic 0546
Industrial 0547
Marine 0794
Materials Science 0548
Mechanical 0743
Metallurgy 0551
Mining 0552
Nuclear 0549
Packaging 0765
Petroleum 0554
Sanitary and Municipal 0790
System Science 0428
Geotechnology 0796
Operations Research 0795
Plastics Technology 0994
Textile Technology 0994

PSYCHOLOGY

General 0621
Behavioral 0384
Clinical 0622
Developmental 0620
Experimental 0623
Industrial 0624
Personality 0625
Physiological 0989
Psychobiology 0349
Psychometrics 0632
Social 0451

The undersigned recommend to the Faculty of Graduate Studies
and Research acceptance of the thesis

“Application of the Medial Axis Transform
in a Design Environment for Die Casting”

submitted by John Michael New, B.Sc.
in partial fulfilment of the requirements for
the degree of Master of Engineering

Thesis Supervisor

Chair, Department of Mechanical and Aerospace Engineering

Carleton University

May 10, 1993
Date

Abstract

This thesis investigates the geometric properties of three dimensional objects known as the distance transform and the medial axis transform, and their use as tools for design, analysis and visualization in the context of an environment for designing die casting dies. Specifically, the distance transform is introduced as an aid to help the die casting die designer visualize where and how the part being cast will solidify, and the medial axis transform is introduced as part of an analysis tool which computes a fast and rough fluid flow field on a simplified representation of the cast object. Descriptions of the uses of these design tools and the methods for implementing them are presented and discussed.

Acknowledgements

I would like to thank everyone in Dr. Goldak's laboratory for their continual support and help throughout my studies at Carleton. In particular, thanks, Naci, for your valuable help, even when you were being pulled in every direction. And thanks, Arun, for your aid and encouragement which helped me see what it was all about. Special thanks go to Dr. Goldak, because you helped me achieve what I came here for, which was to learn.

Table of Contents

	PAGE
Acceptance Sheet	ii
Abstract	iii
Acknowledgements	iv
Table of Contents	v
List of Figures	vii
Chapter I Introduction	1
Chapter II A Review of Die Casting and Die Design	3
2.1. The Die Casting Industry	3
2.2. Die Casting Terminology	6
2.3. Background of Die Casting Hardware	9
2.4. A Simple Description of the Casting Process	11
2.5. Limits of Die Casting	13
2.6. Design of Cast Parts	14
2.7. Design of Dies	15
2.8. Optimizing Design of Dies	17
Chapter III Challenges in Die Design	19
3.1. Use of Computers in the Die Casting Industry	19
3.2. A Typical Design Process	21
3.3. Geometry of Die Cast Parts	23
Chapter IV Some Ideas About a Design Environment	26
4.1. A User Interface for Die Design	26
4.2. Geometric Properties of Objects	34
4.3. A Few Potential Tools.....	37

Chapter V Implementation of the Tools	43
5.1. The Grid	43
5.2. The Distance Transform.....	46
5.3. The Medial Axis Transform.....	51
5.4. Further Processing of the Medial Axis Transform.....	55
5.5. Making the Mesh	56
5.6. Fluid Flow Simulation	57
Chapter VI Conclusion.....	62
6.1. Conclusions	62
6.2. Future Work	63
6.3. Contributions.....	65
References	67

List of Figures

FIGURE	DESCRIPTION	PAGE
Figure 2.1	A multiple cavity die	7
Figure 2.2	The cold chamber and hot chamber injection systems	9
Figure 2.3	The DCRF Nomograph	16
Figure 4.1	Three ways to orient the part in the die.....	36
Figure 4.2 a	The medial axis	39
Figure 4.2 b	Medial axis distances	39
Figure 4.3	Objects and their medial axes	40
Figure 4.4 a	The medial axis of an object with many edges	40
Figure 4.4 b	The medial axis of a smooth-edged object.....	40
Figure 4.5	Three dimensional medial axis surfaces	41
Figure 5.1	A simple two dimensional object in a grid	44
Figure 5.2	Singular cases in initialization of grid.....	45
Figure 5.3	Masks for the two dimensional D-Euclidean algorithm	47
Figure 5.4	Inside and outside of an object.....	49
Figure 5.5	Masks for the three dimensional D-Euclidean algorithm	50
Figure 5.6	DT values of 0.01, 0.25, 0.5, 0.75, and 0.99 of the maximum DT value for a simple object	51
Figure 5.8	Local area for determining if a grid point is on the medial axis	53
Figure 5.9	One quarter disk for radius (5,2)	53
Figure 5.10 a	Part of a lookup table for two dimensions	54
Figure 5.10 b	Part of a lookup table for three dimensions	54
Figure 5.11	Calculated MAT and central MAT region for a simple object	56
Figure 5.12	Results from a simple potential flow problem	59
Figure 5.13	Potential flow on a thin-walled casting.....	61

Chapter I

Introduction

When manufacturers need large numbers of metal parts with high tolerances, good mechanical properties and a good surface finish, at a low unit cost, they often turn to the die casting process to meet their needs. The high pressure die casting industry has been an important part of the manufacturing sector since before 1900, providing the combination of low cost and high quality only when sufficient capital can be invested in the process.

The investment required for die casting is usually great because the machinery is complex, with high demands placed on it in terms of pressures, temperatures, cycle speeds, and running times, and because the die itself is so costly to make and to modify. The performance required of the die makes it expensive to machine and maintain.

In order to help with the design of dies some software analysis tools have been developed. However, the die casting process is extremely complex, and its analysis requires coupling fluid flow, heat transfer, metal solidification, and stress and strain analyses. Furthermore, the geometry of the part or parts being cast can be quite complex, and relatively small features such as rounds and fillets can affect the actual casting process more than might be expected. The geometry of the gating system is also very important since the gating also influences the quality of the casting by quite a large amount. These combined complexities mean that any analysis software that is to be truly useful must be extensive in capability and in capacity. Much of the software that has been developed to date does not have the capability to be used extensively as a tool for die design.

In order for a software tool to be useful, it must not only be able to do the work asked of it, it must also be usable by those who need it. This means an interface must be designed so that the die casting engineer does not also have to be a computer specialist to be able to use the powerful software underneath. This interface will be the focus of this thesis. Emphasis will be placed on design of the gating system, as this is perceived to be the most problematic and least developed art in the science of die design.

Chapter II

A Review of Die Casting and Die Design

2.1. The Die Casting Industry

Some History of Die Casting

Permanent mould casting has been used for thousands of years in one form or another. Gold, silver, lead and bronze were some of the first metals cast in permanent moulds.

Starting about the fifteenth century permanent moulds were used in the printing industry to create type-letters out of lead. This is where pressurization began; a piston and cylinder would be immersed in the molten lead alloy, and the pressure would be used to force the lead into the mould [1].

The first significant use of pressure die casting outside the printing industry was by Charles Babbage, who required the accuracy of a pressurized casting process to make the gears, cams, and other components used in one of his early computing machines [1]. These parts were manufactured with pewter (lead and tin) hardened with zinc. Perhaps Babbage would have approved of using today's workstations to help improve the die casting process.

From Babbage's lead, others started using the process to produce many other small parts, and by 1904 the automotive industry had become the largest user of die casting. In the early 1900's many developments were made in die casting, including using air to supply the pressure (instead of human strength), increased automation of the mechanical process, and the ability to die cast aluminum.

During this century die casting has changed from an undeveloped craft to a mature manufacturing process. It has moved away from being an art and towards becoming a science. However, today mastery of the craft remains a necessary part of the production of a good die casting.

The Industry Today

The process of die casting is one that requires the observance of tradition in many forms. It has such a convolution of mechanisms at work that simply following standards and procedures is not sufficient for success. What die casters have learned over the years is that the deep, inexpressible knowledge required for successful casting comes mainly from experience, with some benefit from the natural aptitude and creativity of the die casting engineer. In essence, die casting is considered by many to be an art.

Standards and procedures set by die casting institutes and national associations, for example, the American Die Casting Institute (ADCI), the ASM Committee on Die Casting, the Society of Die Casting Engineers (SDCE), and the Commonwealth Scientific Industrial Research Organization (CSIRO), play an important role in how die casting companies operate. For a given task at hand, recommended procedures help to define the scope, determine required equipment sizes, capacities and characteristics, and steer the engineer in what is hopefully the right direction with regard to the decision he or she makes. The accepted practices in the industry are like bases upon which other techniques of good design sit. For example, the SDCE provides a formula for calculating filling time based on the metal temperature, the liquidus temperature of the metal, the initial die temperature, and the minimum casting wall thickness [2]. This number is treated as a guide, as there are many factors the die caster must take into account which could affect the filling time.

Most die casting companies have procedures which reflect the particular types of die casting they do. Small companies rely heavily upon a few machines and each machine's idiosyncrasies become known to the operators and engineers. These individual characteristics shape the procedures the company has; often what is written down is very brief, while the real knowledge required for successful castings is held in the experience of the employees and passed on from one person to the next.

Much of what is done in the shop is trial and error because so many factors contribute to every aspect of the success or failure of the casting. Variations in such things as shot pressure, shot speed, temperature, metal composition, and cooling line flow rates contribute a certain degree of unpredictability to the process. Today a greater emphasis is being placed on tighter control over these variables, partly because of stronger market forces demanding better, cheaper castings, and partly because the technology for these controls is available today and becoming cheaper all the time. Today's die casting machines come with much more computer control built-in than those of even ten years ago.

Another major reason for the unpredictability of die casting has been the difficulty in analyzing the process. Until recently it has been virtually impossible to combine accurate calculations of metal flow, heat transfer, solidification, stress and strain in a way that could predict any but the simplest of results related to the process. Without the sophisticated tools, die casters have traditionally relied on charts and tables, along with their experience, as the main tools for solving their casting problems.

With the unprecedented growth of the computer industry, die casting, like just about everything else, has been greatly impacted. The industry has begun to change to meet the needs of global competitiveness. The die casting industry, however, has been slower than most other manufacturing industries to make full use of computers. This is

because of the nature of the industry and the process. When such a large part of a successful die casting process comes from the experience and expertise of the people involved, it is not as easy to include computers as a key part of the process. Thus, the industry has been slower than other industries to make good use of computers as tools for design, analysis, and control.

Despite the power and flexibility of computer systems today, and tomorrow, the experience and plain sense of the die caster will most likely still be needed for the foreseeable future.

2.2. Die Casting Terminology

Die casting is a term used in North America to describe what is called *pressure die casting* in Europe. High pressures are used to inject molten metals into a cavity in a steel die, where the metal freezes and the metal part (or parts) is then removed. Figure 2.1 shows many of the parts of a typical die [3].

Figure 2.1 A multiple cavity die [3]

The die itself is divided into two main halves, the *cover* and the *ejector*. The ejector die half moves away from the cover half to allow the solid part to be removed. The die is designed so that the cast part will stick to the ejector die half, facilitating easy removal of the part. The cover is the half which does not move; the casting metal is fed through the cover half into the cavity.

The channel which feeds the die from the injection system is called the *sprue*. A sprue can be a cylindrical channel or a conical channel. Conical sprues typically have an associated conical *spreader* on the ejector half which serves to accelerate the flow into the die.

From the sprue, the metal goes into a system of channels called *runners*, which transport it to the locations in the die where it will enter the cavity for the part. At the end of a runner's course, its cross-sectional area decreases to increase the velocity further, and the metal then enters a *gate*. Gates are usually short, thin and relatively wide channels which feed the main cavity.

Often the gate is located along the side of the runner and at the end of the runner is a small chamber, a *shock absorber*, which serves not only to absorb the kinetic energy of the initial metal front, but also to trap air and oxides existing in the metal front.

As the metal fills the main cavity which gives shape to the part, the air exits the cavity from thin channels called *vents*. However, often the air and metal travel through *overflows* before reaching these vents. The overflows are placed so the volumes of cooling metal are evenly distributed around the die; in other words, they provide thermal balance to the die as it cools, allowing all main sections of the die to cool at approximately the same rate. They also provide a trap for the oxides and debris at the front of the metal after it passes through the cavity.

Cores are replaceable die parts, often cylinders, required when there is a long hole or other detail in the part. Usually such features require high tolerances, and since this part of the die undergoes greater than normal wear, a replaceable core is incorporated for this purpose. A *slide* is part of the main cavity mould in the die, needed when the part is re-entrant in some way, i.e. the die halves could not separate without damaging the part. A slide is pulled out in a perpendicular direction from the die movement after the part has solidified but before the die opens up.

Inserts are parts of the die which can be changed relatively easily, permitting changes to the casting when required. For example, the company may wish to cast the date into each part, so they could put small numbered inserts into the die and change them every day. Larger inserts are also possible, permitting a large die to be used for many different castings over its lifetime.

In order for the part to be ejected easily from the die, the walls of the die must meet the parting surface at slightly less than ninety degrees. The angle between the walls and the direction of the die movement is called the *draft*. Draft must be included on all walls, runners and cores; the actual draft angle is typically very small; its value depends on the alloy and the height of the wall.

Ejector pins are put into dies to push the part out of the die after it has cooled. These are normally located so that they push on the runner system, but if these are not sufficient, they may be placed at locations where a small mark on the part is not important. They also must be arranged so that they do not damage the casting while pushing it off the ejector die half.

Cooling lines are channels in the die which water or oil flows through to cool the die quickly. They are placed strategically to create uniform cooling around the die, with more lines placed near heavier sections of the casting and fewer near lighter sections.

2.3. Background of Die Casting Hardware

The equipment required for the die-casting process usually includes the two or more die parts, equipment for heating and pouring, accumulators for clamping and filling pressures, part ejection equipment, measurement and control hardware, and auxiliary equipment. Often the machines are very large and are expected to supply high pressures, high plunger speeds, and very high throughputs on an extended basis.

Machines

There are two major kinds of die casting machines, each of which has slightly different processes of metal heating and injection; they are known as hot-chamber and cold-chamber machines (Figure 2.2, from [4]). Hot-chamber die casting uses a pressure cylinder which is immersed in the liquid metal; this process is commonly used for casting magnesium, zinc and lead alloys. The cold-chamber process requires the molten metal to be poured into the shot chamber just before it is injected into the die; aluminum is the primary metal which is used in this process, and copper alloys (for example, brasses) are also cast in cold-chamber machines.

Figure 2.2 The cold chamber and hot chamber injection systems [4]

In hot chamber machines, injection of the metal into the die is well controlled, since the metal volume is regulated by the machine. Cold chamber machines typically require pouring of the metal from a ladle into the shot chamber, and this is more difficult to control precisely. Also, the shot chamber in cold chamber machines contains some air, which complicates the process. Thus, accurate pouring control in a cold chamber machine is an area of some concern to die casters using this process.

The clamping system of the machine is important because it must hold the die halves together with high clamping force while the die is being filled. Clamping pressures from tens to thousands of tons are typical for die casting machines.

High pressure is also required to create the high injection speeds necessary for die casting, and for maintaining the pressure of the liquid metal while the casting is solidifying. This is often achieved by use of a gas-oil accumulator; it accumulates pressure while the casting is being removed and the die is being clamped, then uses the built-up pressure to drive the plunger, forcing the metal into the die.

Ejection of the part from the die is done using pins which push the part out of the cavity. Other equipment is sometimes used to help with the ejection and also the trimming of the part, and transport of the parts away from the casting area can also be automated. The ejection and trimming equipment is important for small cycle times and sometimes to reduce the cost of trimming and transportation.

Dies

The material of the die is important because the life of the die determines much of the unit cost of the casting. Typically a steel like H13 is used, since it has the appropriate high temperature stiffness, toughness and wear resistance for the die casting process. Other metals are sometimes used when the metal being cast is substantially different from the standard die-casting metals.

In many applications the quantities are sufficiently large and the parts are small enough that more than one part is cast in each shot. This means that multiple cavities are machined into the die and the runner system is that much more complex. Multi-cavity dies are quite common in the industry. Figure 2.1 shows a multi-cavity die.

Most dies require cooling lines to help transfer the heat out of the die faster. Some castings require heating in places to balance the cooling rates between different sections of the die, although this is usually not necessary as this balance can often be maintained by placing overflows in appropriate locations.

The die itself undergoes rapid changes in stress and temperature during its use, and the high velocities of the metal inside it tend to cause quite a bit of wear. Thus, designing the die to resist wear and to avoid the possibility of cracking or fracturing is extremely important if a long die life is to be achieved. Typical die lives are in the order of from twenty thousand castings for copper-based alloys such as brass, to one million castings for zinc alloys [5]. Heat treatment of the die is important to die life, and proper lubrication of the die is essential in maintaining a high quality die.

2.4. A Simple Description of the Casting Process

Before the die is closed to begin filling, lubrication is usually added to the die, either by hand or automatically. The amount and kind of lubrication varies depending on the metals of the die and the casting, as well as on many process parameters and casting requirements. After lubrication the die is closed and pressure is applied to keep the two halves together.

The filling of the die is quite fast, typically taking from seven to seventy milliseconds, with some castings taking as little as two and some taking as long as three hundred milliseconds [3]. The liquid metal, usually at a temperature slightly higher than

its melting temperature, enters the die from the feeding system and moves toward the sprue. At the sprue, it spreads out until it reaches the runner. The liquid metal is accelerating all the time as it goes from the shot chamber to the gates. As the metal fluid travels through the runner system, it causes wear on the die. Curved runners are used to reduce this wear, and smooth transitions are made between various parts of the system.

The runners next to the gates are ideally filled before any of the metal goes through the gates. The front wave of metal then fills a shock absorber at the end of the runner, if there is one, while the metal next to the gate is finally forced through the gate by increased pressure in the runner. The velocity of the metal as it enters the cavity is quite high (over thirty five meters per second), and any immediate obstruction to this stream usually undergoes a high rate of wear.

The metal fills the cavity of the part quickly, and then fills the overflows. Once the overflows are filled, the metal begins to flow through the vents, but since the vents are thin and have a large surface area, the metal freezes and the flow stops at this point. This is how the metal flow occurs in an ideal casting situation.

Once the flow is finished, the metal must continue to solidify, and the thermal loads of various parts of the die are balanced using overflows, cooling lines, and occasionally heating, so that each large section of the casting freezes as quickly and consistently as possible. Note that it is virtually impossible to measure exactly when the casting has solidified. Trial and error, as well as rough calculations, are required to ensure the casting is removed as soon as possible after it has solidified.

If the die has any slides, they are pulled out to allow the die to open. The ejector die half pulls away from the cover half after a specified time and ejector pins push the casting off the die. From this point, some systems have automatic removal of the part, or

some kind of initial trimming operation, while some are removed by hand or are just dropped into a bin below the die casting machine.

2.5. Limits of Die Casting

The process of die casting is one of the few processes considered able to produce the final product from its raw material. The limit on dimensional accuracy for small sections is around 0.02 mm (0.0008 inches), which is almost as accurate as machining [5].

Die casting typically produces parts (or sets of parts in a multi-cavity die) at a rate of one to ten parts per minute. Up to five shots per second have been achieved in special cases [3].

Since die casting provides both speed and accuracy, it seems an ideal process for many applications. The main drawbacks are the capital cost of the equipment and the limits on the materials that can be cast. The cost of the process makes it prohibitive for producing a small number of parts. Today, the cost of a die ranges from tens of thousands of dollars to over one million dollars for a large and intricate die. Competition in the industry comes from other casting processes and machining, as well as injection moulding of plastics for some applications.

Defects

If standard gating design practice is not followed or if process parameters such as metal temperature, die temperature, injection speed or pressure, metal composition, impurities in the metal, or improper lubrication deviate from the ideal values, *defects* will usually occur in the cast part.

Misruns are runs in which the die is not completely filled. When two streams of metal meet and they don't melt together (due to oxides, temperatures, etc.) *cold shuts*

occur. *Cracks* and *shrinkage porosity* (internal holes) due to shrinkage of the metal sometimes occur, and *gas porosity* can occur if gases somehow get into the molten metal. *Flow lines* can sometimes be seen on the surface of the part, and this is not acceptable if a good surface finish is important. Other defects affecting surface finish include *galling* (scratching of the surface when the die opens), *blisters* (entrapped gas near the surface of the part), and *die soldering* (when the molten metal solders to the die) [3].

2.6. Design of Cast Parts

When a product is being designed, the process for manufacturing it must be taken into consideration as part of the design process. For die casting this includes keeping in mind the object's size and maximum and minimum wall thicknesses, as well as the necessity for cores, slides, die inserts and part inserts. The flow of the fluid filling the part cavity is often also taken into account in order to ensure a quality product will be the end result.

Each core in a casting increases the cost of the die and potentially the cost of the maintenance also. Still, cores are often required in many castings, and some dies have well over twenty cores.

Slides add quite a lot of complexity to the die and to the casting operation, so they tend to be quite expensive. When designing for die casting, slides are usually avoided if at all possible. Some designs requiring slides can be redesigned so that no slide is necessary, and though this might increase the tooling or add to the weight of the casting, it is often the more economical design.

Keeping the fluid flow in mind when designing parts is important for making the filling process as predictable as possible. Ribs and webs are natural channels for the flow of metal, and if they can be placed so that good filling will result, a good casting is more

likely. Also, if likely or obvious positions of gates are kept in mind while designing, it may be possible to design in good flow characteristics.

These are the kinds of guidelines which come from experience as well as education. The goal in designing a product is almost always to find the least expensive way to produce it with sufficient quality for its application. The roles of die inserts, part inserts, cores and slides are dependent upon the relative costs of achieving that goal, so cost considerations go beyond just casting and into relative costs of other designs and other manufacturing processes.

2.7. Design of Dies

Guides for die design are published by the such groups as the ASM Committee on Die Casting [3], the Society of Die Casting Engineers [2], the American Die Casting Institute, the Die Casting Research Foundation, and the Commonwealth Scientific Industrial Research Organization. One calculation tool which has been widely used in the industry is the DCRF Nomograph (Figure 2.3, from [4]). This tool can now easily be replaced by a calculator or a computer, but it appears to still be in use.

There are certain guidelines die casters use to determine what some of their design and operation parameters will be. Variables such as gate width and height, runner cross-sectional area, and plunger pressure or velocity are normally calculated based on the minimum part thickness, the casting alloy and the part volume. Many theoretical calculations have been done and many experimental observations made, but the ideal process parameters are so dependent on the situation that formal calculations are not accurate enough for every situation; they are used mainly as guidelines for good gating system design.

Figure 2.3 The DCRF Nomograph [4]

The time to fill the cavity is normally determined by the casting alloy and the minimum wall thickness of the part being cast. Smaller wall thickness and alloys with lower melting temperatures require shorter filling times.

With such short filling times, the velocity of the metal at the gates is fairly high, of the order of thirty to fifty meters per second [2]. This number is largely based on the minimum wall thickness and the casting alloy. From the gate velocity the total area of the gates can be calculated. The thickness of the gate is also determined by the minimum wall thickness of the casting, with thinner walls requiring smaller gate thicknesses.

This leaves the width of all the gates to be divided among the different sections of the casting. The number of gates feeding the part is usually based on the geometry of the

part, with each large section having an associated gate. The total gate area is divided so that each gate fills a section of the casting in the time to fill, previously calculated. If there is no division of large sections in the part, the gates are typically laid out along one or more sides of the part so that the total width equals that calculated.

Similar guidelines exist for the size and cross-sectional areas of overflows and shock absorbers, but they are not as critical to the flow as those for the runners and gates. Die casters often cast similar types of parts in different runs, and they use their experience from previous designs to guide them with their present ones.

One of the main uses for overflows is to provide thermal balancing of the entire die. This means that larger overflows are placed near the smaller volume sections of the casting so that these sections do not cool too quickly relative to the larger sections. Often when a part has unbalanced small and large volume sections, die designers use a multi cavity die in which two of these parts are cast simultaneously. In this case they are placed in the die in such a way as to use their symmetry to help create an even heat distribution.

2.8. Optimizing Design of Dies

An optimal die design will be the most cost effective one which produces good quality parts, as quickly and consistently as possible. This means that scrap must be kept to a minimum; the system of runners, gates and overflows are the main areas where scrap can be reduced. Die wear must also be kept to a minimum, for the sake of both the die life and the casting quality. Other cost-intensive design items such as slides must always be compared to the cost of not using them. In die casting there are so many variables to keep in mind that the optimal solution will often necessitate a compromise among all of these variables to produce a good quality, inexpensive part.

As in most engineering designs, a complete analysis of every possible variation in design and its associated cost is not usually possible, but well considered analysis and die casting experience together guide the designer toward the "most acceptable" solution.

Chapter III

Challenges in Die Design

3.1. Use of Computers in the Die Casting Industry

Computers are utilized fully in some areas of die casting today, while in other areas of the industry they are noticeably absent. In designing the part itself, if it is to be die cast usually this means those designing the part belong to a relatively large organization, since die cast parts are almost always produced in large numbers. This means the designers often have state-of-the-art CAD software with which to design the parts. The CAD software is often used in conjunction with analysis software to ensure the part will satisfy its designated functions within whatever constraints are specified for it. Often the designers of parts are not die casters, so their concern with the manufacture of the part is not the same as that of the die casters who will ultimately cast it.

The use of analysis software is also much more common in designing the part than in analyzing the casting process. This is largely because of the availability of reliable software; commercial software for analyzing stresses and strains is much more common and much more accurate than software for analyzing the casting process. Analysis of the die casting process is much more difficult and complex, so accurate and reliable software for it is not yet readily available.

For complex die cast parts, the computer representation is not at all simple. Many parts being designed today still have engineering drawings as their only or their primary representations. These drawings might be on computers in CAD databases, but they are still represented as drawings and they still reflect all the ambiguities and communicative difficulties paper drawings have.

If the representation of the part is only in the form of a drawing, any analysis which is to be done is made more difficult by the need to represent it in a form understood by the analysis software. For example, today making a finite element mesh for a complex part is one of the most manpower-consuming barriers that must be crossed. A very complex part which is represented only on an engineering drawing, even a drawing in a computer, can take several man-years of meticulous calculation and data entry to enter it into a computer in the form of an FEM mesh or an NC machining program.

In the actual die casting sector itself, computers are used most often in the following ways:

- communication of part and die geometry between clients, engineers, technicians and machines,
- inputting data for CNC tooling and EDM processes, for manufacture of the die,
- control and monitoring of the actual casting process, along with auxiliary equipment,
- testing for part and die accuracy, and overall quality control, and
- general corporate management and communication.

Notably, computers are not usually used in the following ways:

- analysis of the die casting process,
- design of the overall die parameters (often calculators are used to determine the basic parameters such as gate width and length),
- design of the sprue, runners, gates, overflows, vents, cooling system, shock absorbers, or

- analysis of the design of the die on the basis of cost, part quality, scrap metal rate, or efficiency of the casting process.

The complex geometry and the difficulty in analyzing the process has, in the past, made computers useless for design except in the most simple respects. This shows the importance of the ability to evaluate a design. Without methods to accurately measure the worth of a design, design software is ineffective. Once software is available that will allow die casters to design the runner systems and to analyze them with some degree of accuracy, they will have to begin using it to stay competitive. Creation of die designs with minimal scrap will be possible, for example, if such software is good enough.

3.2. A Typical Design Process

In a typical die casting shop, design tools are varied and numerous. They include calculators, graphs, charts, manuals, engineering drawings and prototype parts. Several different software packages might be used on a PC or a workstation. The designer is expected to use these tools to design the die, and also to ensure data is transferred from one part of the procedure to the next, doing any necessary conversions or calculations along the way.

The decisions made by the designer are based upon these calculations, his/her experience, and some degree of trial and error. This trial and error part is more pronounced the less experience the die designer has.

As an example of the discussion in the previous session, a typical design process is presented below. This is based upon overall literature information, informal discussions with die casters, and a visit to a die casting and engineering company [6].

Part Design

In many industries, products are designed by looking at an existing product and improving it or changing it to meet new needs or new specifications. What this means is that most changes to the product are incremental; they are minor changes, involving changing some dimension or adding or deleting minor features. Some changes are major, such as combining two different parts from last year's design and making them a single part this year. For many such major changes, many of the old part's features are retained in some form, even though the overall shape or design concept might be quite different. The important point of this is that very few parts are designed from scratch.

If the part is designed in increments, then so are the changes in the process which manufactures the part. In die casting this means gates are most often left where they were before, with the same thicknesses and widths, since they were successful in the past. The thought behind why such decisions were originally made is usually known, for if there is a problem with the present design, a person who has this knowledge will be needed to make changes successfully.

If the part is a totally new design, the cycle of design, testing the casting process, and redesign will be longer than usual. It is here where the experience of the designers of the part, designers of the die, and die casters is most important. Most of the decisions involved with this process are made with the benefit of this experience, and typically with the aid of a calculator and much negotiation between all concerned. No analysis of the casting process is typically done. Important decisions such as how many slides to use are based on a visual inspection of the part and the die designer's experience.

Die Design

Most often, the part is designed by one company and die casting is contracted to another. The design of the gating system is shared by the two companies in various

ratios; some part designers also specify all the gating, some leave it to those who design the die, while most arrangements involve communication and negotiation between the parties involved.

Again, calculators are normally used to determine how much gating is needed; decisions about where to put it, and even how thick the gates should be, are made on the basis of experience with similar parts. Overflows and vents are designed in a similar way, and shock absorbers are only sometimes used, since they are relatively new die casting features. Machine sizes, gate velocities and filling times are also based upon precedent and a few simple calculations.

Feedback

There are times when the design of the part or the die must be changed because either the part is impossible to cast, or the cast part is of unacceptable quality. In these cases, the die caster will discuss the problem with whomever designed the part or die. This feedback is usually an ongoing communication between the parties until all major problems with the casting are solved.

Noticeably absent from the above description is the word "computer." This is because computers do not play a major role in the decisions made by various people. They are used as tools for some of the more mundane parts of the design, but as analysis and feedback devices they are not yet used to any great extent, other than was previously mentioned.

3.3. Geometry of Die Cast Parts

The geometries of parts made by die casting range from fairly simple to very complex. Many of the parameters of a typical part are relatively easy to specify on an

engineering drawing, yet when one tries to represent them with computer software they may become much more difficult to specify. For example, a fillet around the intersection of two cylindrical pipes is simply stated as a fillet with a radius on the drawing, while on a computer a description of the actual surface is necessary, a surface which is not at all simple. Even simple curves, though they may be relatively easy to represent mathematically, pose a problem when an analysis such as the finite element method is used.

Today many parts are designed partially or completely on a computer, using Computer Aided Engineering software, but not all of them. For those that are, each software package has its own way of representing its objects inside the computer. File format converters are commonly used to change files from one format to another. These format converters, however, don't always work, so computer programmers are needed to get them working consistently.

Even if an object is in a format which is understood by a design group, sometimes the geometry must be represented in a different way to be useful to that group. One example of this is in finite element analysis. The representation for FEM is a mesh, which must be created from the knowledge in the previously created part file. There are different representations of meshes within finite element analysis; eight node bricks are common for three dimensional analysis, but there other kinds such as fifteen, twenty, and twenty-seven node bricks, tetrahedra, or triangular prisms. Some software mixes the various element types, while some does not. There are automatic meshing programs available, but they do not necessarily meet the needs of specific programs.

There is some effort being put towards standardization of many of the geometric models in use today, but it may be a long time before such standards are sufficiently useful for most software developers to put into practice. In the non-standardized

environment of today it can be a problem to get commercial geometric modeling tools to work with each other and with privately developed software. In many cases the formats for various geometric models must be changed to accommodate all the software used.

Chapter IV

Some Ideas About a Design Environment

4.1. A User Interface for Die Design

The computer interface is intended to be a tool which helps a die casting engineer, who does not necessarily know a lot about computers, to design dies. This help should take the form of tools to help him/her visualize the design, analyze possible designs, conceptualize alternatives and details of the design, and present the design to those (people or machines) who must use it to create the actual die.

There are three major elements of the interface: design of the die, analysis of potential designs, and visualization and reporting of the results of the analysis. It was pointed out in Chapter 3 how important the analysis is to the overall design process.

The ideal user interface should be easy to use, and functions should be as intuitive as possible. Help should be available on-line for any situation in which the user could get stuck. Visual items should be large enough that the user doesn't have to strain to see them. Saving the work done at any time and retrieving it later should be possible.

Another important aspect of a design environment is the quantity of information the user deals with at a given time. This is called the current context. The computer should provide the user with the parameters necessary for making a few decisions at a time. When several parameters are related such that decisions must be made about all of them together, they should be presented together. One example of this is the "dialog box" found in many windowing environments. The dialog box typically presents a limited number of choices to the user, in the form of buttons, menus, check boxes, or small text fields.

The computer should hide irrelevant information so that the user can concentrate on a small number of parameters at once, and how he/she will deal with them. The parameters not in the current context should not be immediately accessible. Where parameters are very similar and what they mean is very simple, it may be convenient to show more than just a small number of them at one time. For example, a part with twenty dimensions might be easier to deal with if all of the dimensions are shown on one screen, as long as it was clear what each dimension represented on the part. Since the parameters all represent the same thing (distance), in this instance it may not be so confusing to show them all on the same screen.

When a designer is dealing with part dimensions, he/she usually does not need detailed information on the process parameters or the material properties. Thus, these things should be hidden from view in this circumstance. Items which are irrelevant to the present task are not useful; they might confuse the designer while he/she is trying to concentrate on the task at hand. This does not mean that the hidden information is difficult to find; it should be possible at all times to access any of the information available without excess effort, i.e. by changing from the current context to another context.

The design environment should offer a certain amount of flexibility. Where more than one description of a property is useful, it would be valuable to provide each of these descriptions to the user. Since different people think in different ways, each representation of something will be most helpful to a different individual. Of course, the number of ways a process or a parameter can be represented is not usually large for most engineering problems. Still, two views of something are bound to be better than just one, as long as they are consistent.

Ideally, the user should be able to tailor the environment to his/her individual needs. This could also be true for different organizations, since often there are important similarities in most of the products of a given company, and differences in between the products in different companies. Once the company or user tailors the environment to his/her needs it can be used to maximum productivity. Also, this may avoid the frustration of repeatedly having to deal with parameters which are deemed to be irrelevant.

The major elements of a good user interface for die design are described in the next three sections.

Design

The user needs to have tools with which he/she can specify the geometry of the sprue, runner, gates, and other functional elements of a die casting system. These tools should be visually oriented if possible, meaning the designer can see the geometry of his/her overall design at any point in the process.

The tools will also have to include numerical (keyboard) input, since strict dimensions and tolerances are required for most geometrical entities, and mouse-oriented input is not precise enough for this purpose. The numbers used for input should reflect directly the parameters the designer prefers to deal with in his mind. For example, if the cross-sectional area of a shock absorber is the parameter he/she prefers to deal with, he/she should be able to enter the area, not the radius, for this number.

Usually a designer starts with a part he/she wants to make, and tries first to determine the parting surface and then the location of the gates. Once the initial gates are decided, the designer then designs the gate, runner and sprue system which will best produce the desired part.

For determination of the parting surface, the designer typically picks one based on his/her experience and knowledge. It is chosen to minimize the need for cores and especially slides, and so that "good gating" is facilitated. Good gating is itself a complex subject, and the decision of where to put the gates is probably best left to the designer. After analysis the designer might wish to change the gates.

The interface can help the designer define the parting surface and the gates. Using the geometry of the part, the interface could suggest an initial parting surface, and the designer could accept it, change it, or reject it.

At this point it would be useful for the designer to have some indication of the natural flow channels in the part, to help him/her decide where the gates should go. Identifying the sections of the casting with relatively large volumes would also help in this decision. Seeing the likely high flow and high volume areas, the designer could make an initial guess at where the gates and vents should go.

Before going on to design the runners, sprues, shock absorbers, and overflows, it would be useful to have some evaluation of the initial choice of the gates and vents. This would reduce the risk of a fundamental mistake or oversight on the part of the designer in choosing the gates and vents, and to provide faster feedback, which helps speed up the design cycle. Since the location and dimensions of the gates is one of the most sensitive areas of the design of the die, this would be a good place to provide this kind of feedback, even if it were not an exact evaluation of the design.

If this "quick evaluation" is not available, the designer will have to design the runners and sprue based on his/her initial gates and vents, a process which may be quite time consuming, and after analyzing this design he/she will have to adjust the parameters and locations of the gates and vents, which may cause him/her to redesign much of the

runner system. By providing such feedback at an appropriate point in the design cycle, the time and effort required for design could be reduced.

Designing most of the other parts of a die, the sprue, runners, shock absorbers, and overflows, is primarily a problem of specifying the desired geometries and geometrical relationships of these entities. Guidelines mentioned in Chapter II are typically used to select numbers for these parameters.

What the interface should provide at this point is the necessary design tools to enable these guidelines to be followed. For example, the DCRF nomograph could be provided in equation form so that the designer can easily use the calculations and resulting information the nomograph provides. Another example of one of these design tools could be one which shows the cross-sectional area of the channels the liquid metal is expected to flow through. This would allow the designer to ensure that the area does not increase suddenly at any point along the route. All relevant tools of this nature should be a part of the interface if possible.

Specification of cooling lines could be quite complex, since there are several possible geometry designs and several parameters associated with cooling. The user should be able to specify where the cooling lines are and what geometry they have. The flow rate or the cooling rate in each should be input, depending on which the user requires.

The user interface should allow selection of the casting and die materials, and perhaps some of their properties such as composition. The ideal environment will contain a database of all the potential materials to be used by the die caster, and some mechanism for selecting, checking and editing the materials and their properties.

Some process parameters also need to be specified as part of the design. Flow rate as a function of time is important in order for proper filling to occur. Ambient

pressure and temperature are important for accurate analysis. Cooling rate, or flow rate in the cooling lines, must also be specified. These parameters tie in closely with the analysis, and they will be briefly discussed in the following section.

Analysis

In this thesis, the analysis portion of the software can be done largely automatically using finite element analysis and other software. All the necessary boundary conditions and initial conditions can be set by the software, based upon the parameters input in the design section.

For the analysis to provide the best possible information to the user, a combined analysis of fluid flow, free surface propagation, solidification, stress and strain should be run. Since this analysis will take several hours or days (using the software being developed today), the user would prefer to be able to do an analysis with only one or two of the above kinds of analysis, and this capability should be provided.

The analysis should be able to predict many of the common defects which occur in the casting; it should be possible to see misruns, cracks and shrinkage porosity if they occur in the part, and the nature and extent of such defects should be visible. Whether or not it is possible to predict cold shuts or any of the surface finish defects which occur in reality is not known at this point, but it is expected that some time in the future they will be calculable. Since defects depend on how the fluid flows through the system and how heat is removed from the part and the runner system, an analysis of fluid flow and heat transfer is expected to be useful.

The analysis software should also be sufficiently flexible to allow different types of input parameters like pressures and temperatures. Vacuum casting should be possible. In real die casting, often the shot speed has three distinct pressure profiles and flow rates; one to fill the die up to the gates, one to fill the die completely, and one to pressurize the

metal as it solidifies. This kind of detail should be possible with the interface. Other parameters such as material properties of the die and the cast metal should be sufficiently flexible to allow some variation.

In order to optimize design of the die, it would be desirable if the designer could examine the effects of slight or large variations of one or more parameters. The analysis portion of the interface should allow this kind of sensitivity analysis where possible. For example, to examine the final freezing of the metal, the user might wish to simulate the filling of the die once, then simulate what happens just after the die has filled many times using several different dwell pressures (pressures during solidification), and/or cooling line flow rates.

The accuracy of the analysis depends on several factors. One is the accuracy of the FEM mesh used. Another is the capability of the software in modelling the process. Also, the quality of the input data is important. It is not known how accurate the software being developed today is, but it is clear that modelling of the casting process will become more accurate as more research is done over the next few years.

In an ideal design environment, of course, the accuracy of the software would be excellent, and this would allow the designer to design dies which worked on the first attempt. Even if the simulation is not perfect, however, the process of designing the die could be greatly improved with the analysis software.

In practice, since the geometry and the process are so complex, the computational costs of doing the analysis will be somewhat expensive. What this means today is that a complete analysis as mentioned above, on a complex part, could take several days or weeks on a typical workstation, including data entry and cpu time. The computers of tomorrow will no doubt be much faster and just as inexpensive, so this time will decrease

dramatically over the next few years, and the hardware costs will remain small to moderate.

Once the casting process has been simulated, the designer will often have to go back to the die design part of the software. This then becomes a loop of design, analysis, redesign, etc. The analysis software should provide any information possible to aid in redesigning the gating system. If the time to complete the final design can be shortened by some early feedback mechanism (or by any method), it would be of great value to die casting companies.

Visualization

One topic that has not yet been discussed is how the results of the analysis could be presented to the die designer. This presentation is very important because it is a vital part of the feedback mechanism the designer requires.

It would be useful for the designer to be able to see any of the parameters he/she needs to see. For die casting this means visualization of fields such as the following:

- location, velocity, and acceleration of the liquid/air interface
- temperatures
- pressures and velocities
- location of the solid/liquid interface during freezing
- heat flux
- stress / strain.

All of these calculated data fields should be visible at any point in the casting and at any point in time of the analysis.

Standard viewing methods such as viewing these fields on a cutting plane, viewing them as they change in time, showing only the remaining liquid, viewing from different angles, and looking at different sections of the casting independently, would be

useful in letting the designer see exactly what he/she needs to see. Graphical workstations today are capable of showing these kinds of views, and there is even the possibility of viewing in three dimensions (stereoscopically) if it is useful.

The ability to write output parameters into various report formats, graphs and presentation slides, would be useful, but perhaps it would be more appropriate to use a user-friendly, powerful presentation graphics program. This would only require the casting interface to write this data into a format which is readable by such a presentation graphics program.

4.2. Geometric Properties of Objects

This section poses the question, “Are there geometric properties of objects which we humans can recognize, but which are not easily represented on a computer?” The reason this question is asked is because if its answer is yes (or even a qualified yes), we can ask another, related question, “Is there any way to represent and use these properties to help the designer design the die, and if so, how?” This is one of the fundamental questions of this thesis, and this section is dedicated to exploring the question and perhaps guessing at some possible answers.

When one looks at an object, one sees an overall shape as well as several details. Often one sees functional aspects as being the most important, if the object is a functional object. Often the useful parameters of an object are so fundamentally important that we do not notice them, expecting them to be there in every instance of the object; we do notice, however, when they are absent. When one looks at a chair, one might notice the design of the chair, or its color, but not the fact that it has four legs. If the chair were missing a leg, however, one would normally notice the absence as a fundamental flaw in the chair. If an artist looked at such a piece of furniture, he/she might still note the design

or coloring with more interest than the missing leg. So really, the perspective of the person regarding an object must play a role in what kinds of parameters are noticed.

When a die designer looks at a part to be cast, no doubt he/she sees such things as parting lines, holes, reentrant sections, large volumetric sections, and natural flow channels, if any of these exist and are immediately obvious. Perhaps such parameters as "flatness," or "roundness," are noticed, or that flat sections are parallel or perpendicular, or that they are not. When an engineer sees a three metre rod, he/she immediately knows that it can not be die cast at all (or technically, not without extreme expense).

Is it possible to represent any or all of these features in a software model? How can we possibly measure the "obviousness" of such geometric entities?

One possible way to proceed would be to consider some properties which are known to be calculable. Center of gravity and moments of inertia are two quantities which can be calculated relatively simply if the geometry is known. One measure of the "flatness", mentioned above, could be the maximum moment of inertia. For many flat objects the axis of the maximum moment of inertia could also be the direction of movement of the die halves.

The most useful kinds of properties, for our purposes here, are those that would aid in the design of the die. If there are no re-entrant sections in the part with respect to a particular direction or axis, it would mean this axis could be placed in the direction the die halves move, and no slides would be required. This can be seen as a sort of "vertical convexity," and there is some literature on a subject related to this [7]. This kind of aid would be more useful in some situations than the maximum moment of inertia, for this vertical convexity is exactly what is required for dies with no slides.

Comparing the examples in Figure 4.1, the first choice for placing the object in the die might be to put the large flat face perpendicular to the die movement direction;

Figure 4.1 Three ways to orient the part in the die

this results in interference of the left rib with the die when the part is ejected. Another choice for placement of the part might use the direction of the principal moment of inertia; this still creates interference during part ejection. If the orientation for which the part has vertical convexity is chosen, the problem of interference is no longer encountered. Note that in this last case, the parting line of the die might be more complex, so perhaps the cost of a die with a more complex parting line and no slides must be compared to the cost of a die with a simpler parting line but which does contain a slide.

This vertical convexity is not considered further in this thesis, except to state that it would appear to be a useful tool for die design software, and further investigation of it might be warranted.

Other tools which might be useful would be tools to identify large volumetric sections of an object, and to show natural flow channels in an object. These properties could be used by the designer to design gates, vents, and overflows, which are a few of the more sensitive areas of the design.

4.3. A Few Potential Tools

The Danielsson Distance Transform

One way for the computer to determine if there are large volumetric sections in an object is to go to each point inside the object and calculate the minimum distance from the point to the surface of the object. This will give a mapping of points to minimum distances from the surface. Calculation of this minimum distance at every point in the object is not a simple matter.

In digital image processing a similar problem has been studied for several years, only in two dimensions. In these exercises, two dimensional objects are discretized and represented graphically and mathematically by a regular array of pixels or points. The minimum distance from each of these pixels to the edge of the object can be calculated, and the set of points and their related distances is called a distance map or a distance transform (or DT).

Danielsson [8] showed how such a distance transform could be calculated sequentially, with true Euclidean distances, at a cost proportional to the number of points. His first algorithm was shown to have small maximum error, while the error in his second algorithm was negligible.

Borgefors [9] showed how the Danielsson algorithm can be directly extended into three dimensions, which is useful here. In three dimensions there is a significantly larger number of points. Both the two dimensional and three dimensional algorithms will be described later in more detail.

Visualization of the DT in three dimensions can be done in various ways. Colors could be used to represent distances, and each point could be displayed with its appropriate color. Another method could be to show only those points whose DT values (i.e., distances) were greater than some threshold value; the magnitude of this value could

be determined by the designer. A combination of these two methods might also be useful, showing only color coded points whose values are above the threshold value.

This threshold method could even be seen as a crude sort of solidification analysis; normally it is the thickest sections of a casting which solidify last, and these sections contain exactly the points with the highest DT values. For example, sequentially showing the points in order of their DT values in groups of 0.0 to 0.1 of the maximum DT value, then 0.1 to 0.2 of the maximum, etc., could be interpreted as a first approximation of how the object will solidify. This method would not be accurate enough to determine exactly where defects would occur, but as a tool for the designer it might be useful. The designer could use it as a general tool to help him/her keep in mind where the thick sections are, their overall shapes, and to accurately tell how thick such sections are.

Medial Axis Transform

Flow of the liquid metal in the cavity is another phenomenon the designer would like to be able to predict easily. Some castings have natural flow channels as part of their shape; for example, ribs and webs might serve as the initial channels which direct the fluid through the die cavity. Other castings, however, are more complex, and it might be difficult to guess with adequate or useful accuracy the flow patterns of the metal as it fills the cavity.

The distance transform of Danielsson mentioned earlier might be of some use here. The distance transform's value at the intersection of a rib and the flat section it lies on will be slightly greater than the value at the flat section itself. Thus, a line propagates along the rib where these local maxima occur. Danielsson in fact discusses these local maxima, again in two dimensions, and gives an algorithm for finding the set of points which correspond to all of the local maxima in his distance map [8]. This set of points is

called a skeleton or a medial axis. The set of points and their corresponding distance transform values is called a medial axis transform (MAT). The algorithm can be expanded to three dimensions in a straightforward way, although no literature on this transformation was found.

The medial axis transform in two dimensions has some interesting properties. The points on at least two opposite sides of a medial axis point have lower DT values than the point itself, and each of these point's distance to the edge of the object corresponds to a different edge. This means that the medial axis point itself must be the same distance from at least two edges. The MAT appears as a line (or curve) which is centered between these two edges (Figure 4.2 a).

Figure 4.2 a The medial axis

Figure 4.2 b Medial axis distances

Where three edges are the same distance from a point there can not be a curve or a line, just a point (Figure 4.2 b). Each pair of edges has its own section of skeleton, and these sections meet at a point where three or more edges of the object are all the same distance from the point.

The MAT of a complex shape, such as in Figure 4.3, on the right, shows some of the kinds of things one might expect in most medial axis transforms. Two main types of skeleton appear to exist in such a picture: one which is the center line between two adjacent edges, and one which is the central axis between two opposite edges.

Figure 4.3 Objects and their medial axes

The former appears to be a sort of boundary axis, while the latter looks like a central spine of the object. This distinction can be seen in Figure 4.4 a, which has many straight edges approximating a few curves; the boundary medial axes are numerous, and do not tell us a lot about the shape of the object, even in its local area, while the central medial axes seem to form a spine and combine to show an overall shape. If the object were in fact made up of the curved edges (Figure 4.4 b), its MAT would include this central part, but the boundary MAT sections would not exist.

Figure 4.4 a The medial axis of an object with many edges

Figure 4.4 b The medial axis of a smooth-edged object

In discussing the medial axis transform it has been convenient to refer to it as a set of lines or curves, while actually it is a set of points. The points, however, clearly fall upon the lines, and with sufficient resolution, for the purposes of this thesis, it is convenient and not unsuitable to treat the point sets as lines or curves.

Figure 4.5 Three dimensional medial axis surfaces

In three dimensions, the MAT is a set of points which describe a surface. The points are those which bisect two external surfaces of the object. Again there can exist boundary MAT sections and center MAT sections, both of which can be represented as surfaces. With the added dimension, it is found that where two MAT surfaces intersect there is a line or a curve, and where two or more of these lines intersect there is a point (Figure 4.5). Each point on a line is an equal distance from three external surfaces, and each intersection of such lines is an equal

distance from at least four external surfaces. If the boundary MAT sections (those whose external surfaces are adjacent) are taken away, the resulting center sections again define a surface which can be interpreted as a sort of flat representation of the original object.

If one takes just the center medial axis of a two dimensional object, it appears to be a very much simplified version, or a stick-drawing, of that object. The center medial

axis of a three dimensional object seems to be a flat model of the original object. What is perhaps most interesting about this is that this stick-drawing and this flat model can be calculated on a computer. How these calculations can be done will be discussed in the next chapter.

Chapter V

Implementation of the Tools

It was mentioned in the previous chapter that if a fast, basic fluid flow analysis could be done during the initial design of the die, this step and thus the overall design process might be improved. It was also mentioned that the central part of the medial axis transform appeared to be a simplified representation of the part. If these two ideas are used together, perhaps a simple fluid flow analysis could be done on the medial axis representation of the part. Since the MAT is a set of points on a surface, this would in effect be a two dimensional analysis in three-space. This two dimensional analogy to a fluid flow analysis could be much faster than the three dimensional version.

Starting from this assumption, a two dimensional fluid flow analysis has been done. Details of how it was done follow in the next sections, followed by some results and discussions in the next chapter.

5.1. The Grid

Danielsson's distance transform in two dimensions will be discussed in detail in the following few sections, followed by the addition of the third dimension, which is a straightforward extension [8, 9]. All of these algorithms have been implemented in Smalltalk-80, version 4.0, and it is expected they can be changed to C or C++ without difficulty so that their speed is improved.

In order to use Danielsson's distance transform in the casting interface, an array or grid of points must be created. This grid is based on the boundary of the part and on the maximum computational cost allowed. It will be assumed here the boundary of the object is known.

A bounding box of the object is first calculated, with the axes of the box (a rectangle) along the x and y dimensions. This box is then used to create a regular grid of points, which is expanded by one on all four sides of the box to ensure the object is enclosed within the grid. Each point has a corresponding distance function value, which is represented as a pair of integers. This value is not the same as the location of the point, which must also be known. The distance function values are all initialized to (0,0). For notational purposes, sets of points with common y coordinates will be called rows, while sets of points with common x coordinates will be called columns (see Figure 5.1).

The grid must next be initialized to the format required by the DT algorithm. What this means is that a large value must be stored in the DT value of each point inside the object, while the points outside the object will retain the value (0,0). With a large grid, this initialization will take some time, but it can be done in $O(n)$ time, where n is the total number of grid points.

Figure 5.1 A simple two dimensional object in a grid

The initialization algorithm follows here, described in words. It was written for a boundary consisting of straight edges, represented by pairs of endpoints. It is expected that similar code could be written for other boundary representations, but it seems likely that code for curved surface representations would be more difficult to write.

- for each row of the grid, except the first and the last, do the following:
 - find the intersections of the line, whose equation is $y = y_n$, with the boundary of the object; there should be an equal number of such intersections; this intersection is handled in the following manner:
 - for each edge of the boundary;
 - check to see if the edge's pair of end points are both above or both below the line; ignore all such edges.
 - check to see if both points are on the line; ignore all such edges.
 - check to see if one point is above and one point is below the line; if this is the case, find the intersection of this edge with the line and store it.
 - check to see if one point is on the line and one point is below it; if this is the case, store the point which is on the line.
 - check to see if one point is on the line and one point is above it; ignore all such edges.
 - the purpose of distinguishing the six cases above and rejecting four of them was to properly deal with singular cases such as those in Figure 5.2; in case A, no intersection is possible; in case B, an intersection is required; cases C to G represent the possible singular cases, which could cause problems; in case C, the point on the line is stored once, since the edge below it is

Figure 5.2 Singular cases in initialization of grid

accepted and the edge above it is rejected; in case D, the point on the line is stored twice, since both edges below it are accepted; in case E, each point on the line is accepted once, since each belongs to an edge below the line, which is accepted, and each also belongs to an edge on the line, which is rejected; in case F, the point on the line is not stored, since both edges above it are rejected; in case G, no point on the line is stored, since none of the edges have endpoints below the line; in case H, the left point on the line is stored once, since only one edge is below the line.

- sort each of the intersections (stored points) from lowest to highest x coordinate value; if a point occurs more than once, remove all instances of that point, unless it occurs an odd number of times; in that case leave one instance of the point in the list.
- call the first grid point an outside point; go through the grid points in order, determining how many of the intersections have a smaller x coordinate value; if the number is even, call the grid point an outside point, while if the number is odd, call it an inside point.
- give each inside point a distance transform value of (M,M) , where M is an integer greater than the number of rows in the grid.

This method for initializing the grid could no doubt be improved upon. It was done this way only because it was a simple yet effective method. In Danielsson's paper, the initialized grid was assumed given, but it is seen here that it is not a trivial problem.

5.2. The Distance Transform

Danielsson [8] calls the more accurate of his two methods for creating a distance transform "eight-point sequential Euclidean distance mapping." It is named thus because

each point checks each of its eight neighbors in determining what its distance function value will be. Figure 5.3 [9], shows the basic steps of the algorithm.

Starting at the second column and the second row of the grid, each row in order is traversed forward and then backward once; then in reverse order, each row is traversed backward and then forward once. Comparisons are made with the neighboring points, and these comparisons lead to the final distance function value of each point.

forward pass:

backward pass:

Figure 5.3 Masks for the two dimensional D-Euclidean Algorithm [from Borgeltors [9]]

The four-point algorithm is the same except it does not include the corner neighbors (in Figure 5.3, the $+(1,1)$ values).

The algorithm is briefly described:

- for each row except the first, in order, do the following:
 - for each point except the first, in order, compare its value with the values of four of its neighbors; the comparison is made by adding $(0,1)$ to its neighbor above, $(1,0)$ to its left neighbor, and adding $(1,1)$ to its two diagonal neighbors in the row above it; compare the sum of the squares of the distance function values, as this represents a comparison of Euclidean distances;

store at the grid point the distance function value corresponding to the smallest of these compared values.

- for each point in the row except the last, in reverse order, compare its value with that of its right neighbor plus (1,0), using the same comparison method as above, and store the minimum value.
- for each row except the last, in reverse order, do the same procedure as above, only using the opposite directions, as indicated in Figure 5.3.

This algorithm results in a distance map with very small error (0.09 of a pixel, for any size grid) [8].

Conversion of this algorithm to three dimensions is not difficult. Borgefors, in his paper, "Distance Transformations in Arbitrary Dimensions," presented detailed algorithms for distance transforms in one, two, three, four, and arbitrary numbers of dimensions [9].

For notational purposes, sets of grid points with common z coordinates shall be called levels. Each of the grid points, before initialization, are given the value (0,0,0). Initialization then occurs in much the same way as in two dimensions. The object is represented by its boundary in three dimensions.

To initialize the grid, at each level, the intersection of the object's boundary and the plane $z = z_n$ is found. The level is then treated exactly as a two dimensional grid for purposes of initialization. The intersection is a two dimensional object, again represented by its boundary, and it can include more than one connected, closed object. It can also include a closed object inside another closed object; in this case the initialization algorithm treats the space inside both objects as outside. In fact, if a point on the grid is

surrounded by an even number of objects, it is considered an outside point, while if it is surrounded by an odd number of objects, it is considered an inside point (see Figure 5.4).

Figure 5.4 Inside and outside of an object

Computing the distance transform is still an $O(n)$ operation, where n is the number of grid points, but n can be quite large, so the computing time can become quite large. For example, in the code implemented here in Smalltalk-80, a 20 by 30 by 80 grid took several hours to calculate on a typical workstation, although significant reduction in this time could be possible with code review and optimization.

The three dimensional algorithm is very similar to the two dimensional one. Each level is operated on in a forward pass, then a backward pass is done. On each pass, each level is treated much as the two dimensional distance transform, except it first checks nine points in the level just before it. The nine points checked are the adjacent point, the four diagonal points, and the four corner diagonal points. Figure 5.5 is a description of the overall process.

Figure 5.6 DT values of 0.01, 0.25, 0.5, 0.75, and 0.99 of the maximum DT value for a simple object

5.3. The Medial Axis Transform

A description of the medial axis transform will be given here for a two dimensional grid, since it is easier to follow than the three dimensional version, which will be described afterward. The medial axis is often thought of as a skeleton of an object, or as the set of local maxima of a distance transform. This latter definition is one which Danielsson states, but the points are not local maxima in the strict sense; they are points where the gradient of the distance function switches direction. He uses another definition which is important for understanding the algorithm he uses to go about calculating the MAT; according to Danielsson [8], the skeleton "consists of those points (i,j) for which maximum-sized disks

$$D(i,j) \in S$$

can be found that are not completely covered by any other (larger) disk" (S is the set of grid points).

By a disk he means a circle with radius equal to the magnitude of the distance value of a point, and all the points inside that circle. Such a disk can be created by calculating the distance map of a circle of the stated radius. An example of one quarter of one of these disks is shown in Figure 5.7.

Figure 5.7 One quarter of the disk for radius (4,2)

The medial axis transform is computed from the distance function by considering the neighborhood of each grid point, excluding the points with values of (0,0). Each of the grid point's neighbors has a distance value which in turn has a corresponding disk whose radius is the magnitude of the two-numbered value, i.e., $\sqrt{i^2 + j^2}$. Each grid point checks each of its neighbors to see the neighbor's disk completely covers its own disk. If none of its neighbor's disks completely covers its own disk, the grid point is a member of the medial axis transform.

The disk of any neighbor can not enclose (i.e., go beyond) the disk of the grid point being tested because the distance function algorithm will never create that situation, since the distance function value of any neighbor must be no greater than that of the point, plus (1,1). The two remaining alternatives, namely just covering the disk and not completely covering the disk, are the cases which determine whether a grid point is in the medial axis or not.

The way to determine if a neighbor's disk covers its own disk is best described using an example: looking at Figure 5.8, if the point being tested to see if it is on the medial axis is point A, whose distance map value is (4,2), it has eight neighbors, b, c, d,

e, f, g, h and i. First, neighbor b is checked; b's value is (5,2), and this value corresponds to a disk of radius $\sqrt{5^2 + 2^2}$; this disk is shown in Figure 5.9.

Since b is a diagonal neighbor, it checks its disk's point diagonal from the centre, which is (4,1). This number is compared to point A's value, (4,2) and it does not have the same magnitude. If the two values had the same distance function magnitude, then b's disk would just cover A's disk, and A could not be on the medial axis. Since the two values do not match, point c (also value (5,2)) is then

b (5,2)	c (5,2)	d (5,2)
e (4,2)	A (4,2)	f (4,2)
g (3,1)	h (3,2)	i (3,2)

Figure 5.8 Local area for determining if a grid point is on the medial axis

Figure 5.9 One quarter disk for radius (5,2)

checked. Point c's disk is the same as point b's, Figure 5.9, and since it is a direct neighbor, it reads the value (2,4), the point directly next to the center value. Since (2,4) and (4,2) have the same magnitude, it has been found that the disk of point c just covers the disk of point A, so A can not be on the medial axis. The only time a point is found to be on the medial axis is when none of the eight neighbors has a disk which covers A's disk.

The method for finding the medial axis is simply to go to each grid point, perform the above test, and collect all the points whose disks are not covered by any of their neighbors' disks.

Since this method requires two values from a distance map of the disk of each neighbor being tested, the distance map of each disk must be calculated. This can be done once for each possible disk radius and the two values required by the test can be stored in a look up table, bypassing the need for calculating many distance maps every time a medial axis transform is calculated (see Figure 5.10 a).

radius value	direct neighbour	corner neighbour
4, 2	2, 3	3, 1
5, 2	2, 4	4, 1

Figure 5.10 a Part of a lookup table for two dimensions

radius value	direct neighbour	diagonal neighbour	corner neighbour
5, 3, 0	5, 2, 0	4, 2, 0	4, 1, 1
5, 3, 1	5, 2, 1	4, 2, 0	4, 1, 0

Figure 5.10 b Part of a lookup table for three dimensions

Implementation of this in three dimensions is again fairly straightforward. Instead of disks, spheres are used, and each grid point checks its twenty-six neighboring points to determine if it is on the medial axis. There are two potential problems with the lookup table. The first is that each sphere must be calculated in order to get the values to put into the lookup table, and for large spheres, this can take a long time when many spheres must be created. The second problem is that the lookup table itself may have to be very large. For example, for a potential grid size of 100 by 100 by 100, the maximum distance map magnitude is 50, and the number of possible spheres with radius less than fifty is very large. In the lookup table, nine integers must be stored for each radius (Figure 5.10 b). If some way can be found to calculate these numbers quickly, rather than store them, a great deal of memory and initial computing time could be saved; as yet no such method has been found.

5.4. Further Processing of the Medial Axis Transform

Since the distance transform and the medial axis transform in three dimensions have been discussed sufficiently up to here, the remaining discussion will be restricted to three dimensional objects. The three-D object is represented for these purposes as a finite element mesh, with quadrilateral external faces which share edges and nodes. Other representations are possible; still much of the discussion will hold for many of these other forms.

In Chapter 4 it was mentioned that two subgroups of medial axis sections appear to exist, boundary sections and center sections. The algorithm for creating the medial axis does not distinguish between the two types. This distinguishing trait seems to be strongly related to how many and which external faces of the object are the same minimum distance away from the given grid point.

If one wanted to remove the boundary MAT grid points from the collection of MAT points, one would have to find all of the MAT points which were the same distance from exactly two adjacent external faces, where "adjacent" means sharing a common edge. This is possible in practice, and it has been done with reasonable results.

The method for calculating this is as follows. For each point on the MAT, find the faces which are a distance away equal to the DT value of the point, plus or minus the distance from a grid point to its nearest neighbor. If and only if there are exactly two such faces, and they are adjacent, then the grid point is a boundary MAT point and can be excluded from the central MAT list of grid points. Figure 5.11 shows the medial axis points and the calculated central MAT points for a simple object.

Figure 5.11 Calculated MAT and central MAT region for a simple object

Finding distances to all the faces may take a fair amount of time, but it should be kept in mind that the MAT has a small fraction of all the grid points in it. Furthermore, with the availability of advanced geometry functions it should be possible to optimize this function to take less time.

If, after determining which faces are associated with each MAT point, pointers to each face are stored for each point, this information can perhaps be used later to sort the central MAT points into groups.

5.5. Making the Mesh

Assuming we have successfully calculated the set of central MAT points, the next step is to create a finite element mesh so that it can be used for analysis. There are two main possibilities for constructing the mesh, automatically and manually.

Creating the mesh automatically could involve using the central MAT points to define a group of surfaces. These surfaces could then be used as a basis for finite elements; curvature at any point could be used to plan how fine the mesh should be, and the MAT points which are associated with three or more external faces could be used as guides for where the various surfaces intersect. Automatically computing the mesh has

not been studied in detail at this point, so further work must be done to determine how difficult it would be to implement.

Creating the mesh manually is the method used for the work done on this thesis. The central MAT points are displayed on the monitor and the points are connected using the mouse. While creating the mesh, the set of MAT points can be rotated or zoomed in upon in order to facilitate picking the points with the mouse. In this method it is up to the person creating the mesh to ensure that it is constructed properly. In the system used here, every element of the mesh must be a four node quadrilateral element. The nodes of the elements are located exactly where the MAT points are, so they have x , y and z coordinates in the same scale as the original object. Although the resolution provided by the grid points is not high, it is deemed sufficient especially when considering the accuracy of the analysis for which the mesh will be used.

5.6. Fluid Flow Simulation

The quick fluid flow analysis to be done is a two-dimensional potential flow. Even though the mesh exists in three dimensions, the analysis is no different if the mesh is flattened to exist on a plane as long as the topology of the mesh and the geometry of the elements are preserved. Thus, a two-dimensional analysis is what really is being done.

Normally potential flow is used as a tool for high Reynolds' number flow, where forces from viscosity effects are far outweighed by inertial forces [10]. It is recognized that the conditions required for potential flow are not necessarily satisfied in real castings. Since this simulation is intended as a rough first approximation, no attempt will be made to quantitatively verify any of the results. Metal flow in die casting is an unsteady flow process with many variables which must be taken into account if an accurate flow

analysis is to be done. It was mentioned before that a more accurate analysis should be done at a later stage in the die design process, after a few of these “quick analysis” simulations are done and a satisfactory die has been designed according to the initial rough results.

Potential flow is defined by the equation

$$\mathbf{v} = \nabla p$$

where $\mathbf{v}$ is the velocity vector and p is the velocity potential, which can be defined by this equation. This equation holds only for irrotational, inviscid, incompressible flow.

The FEM formulation of the problem of two dimensional potential flow is described in Cook, Malkus, Plesha [11]. Boundary conditions for the problem are required for two kinds of boundary, one which represents the inlets (gates) and outlets (vents or overflows) for the fluid, and one which represents the edge of the object. At the inlets and outlets, values of potential are prescribed. In the example done in this thesis, the inlets were set to 200 and the outlets were set to 0 (the actual numbers do not really matter; 1.0 and 0.0 would also have been adequate). On the remainder of the mesh boundary the gradient of the potential (the velocity) is set to zero perpendicular to the boundary. Thus, on the boundary, every node is prescribed exactly one condition, either a velocity component or a velocity potential value.

It would be equally possible, and perhaps more intuitive, to prescribe a velocity at the inlet instead of a potential. The reason this was not done in this thesis is that the actual FEM program used was developed for heat transfer, where it is more convenient to prescribe temperatures than velocity potentials.

The FEM formulation of this heat transfer problem is also described in [11]. The potential flow problem is analogous to the heat transfer problem, where the velocity potential is replaced by temperature and velocity is replaced by temperature gradient.

M. Gu describes how surface elements (called shell elements) in three dimensions are transformed to two dimensions using a mapping from global to local Cartesian coordinates [12].

The analogy between heat transfer and potential flow can be considered to be due to the identical mathematics of the two problems. In steady state heat transfer, if no internal heat is generated or dissipated,

$$\nabla \cdot k \nabla T = 0$$

where k is thermal conductivity and T is temperature. Similarly, in potential flow, if there are no internal sinks or sources,

$$\nabla \cdot \nabla p = 0$$

where $v = \nabla p$ as described earlier.

In this thesis, the method is demonstrated for potential flow on a flat surface though the code was written for three dimensional quadrilateral shell elements (Figure 5.12).

Figure 5.12 Results from a simple potential flow problem

For each node on each element, the velocity was calculated by using the values of the potential of the node and its two neighbors on the element. The three nodes define a plane, and their three potentials define a linear potential function within that plane. The gradient of this linear function was then calculated, and this was treated as the contribution of the nodal velocity from that element. Thus, for each node, one velocity was calculated for each element it was a part of, and these velocities were averaged to get the velocity at that node. There are other ways to calculate the nodal velocities, and it would be possible to calculate an elemental velocity too; this method was used because it was a simple way to illustrate the velocity field in the mesh.

Looking at Figure 5.12, it can be seen that the velocities on the boundary nodes are not parallel to the boundaries. This is probably due to the coarse nature of the mesh, and the method of calculating velocities.

Usually the designer has a desired gate velocity, and the average of the calculated relative velocities at the inlets (gates) can be multiplied by some number to achieve this desired gate velocity. This number is simply the desired gate velocity divided by the average calculated gate velocity. The relative velocities at all points can then be multiplied by this number to get the actual velocities everywhere on the mesh.

As a final example, Figure 5.13 shows the nodal velocity field for a simple box shape. It indicates that the nodal velocities can be calculated in three dimensions, although they are not in the planes of their elements when their elements meet at sharp angles. At the four parts of the box which jut out, the velocities appear quite errant; this is probably due to the coarseness of the mesh as well as the angle at which the elements meet.

Figure 5.13 Potential flow on a thin-walled casting

In this chapter it has been shown how, given a boundary representation of a part, a velocity field can be calculated. In addition, an indication of thicknesses of various sections of the casting is calculated during the distance mapping stages of this overall process.

Most of the process is done automatically by the computer system; only the meshing of the flat representation and the selection of gate and vent/overflow locations are done manually at this point. It appears possible that the meshing could be done automatically also. This would leave only the gate and vent/overflow selection to the designer, which is the part of the process we wish the designer to have control over.

Chapter VI

Conclusion

6.1. Conclusions

In Chapter 3 it was explained how today's die casting organizations make little use of the powerful tools available to them. It was suggested that a better "design environment" could be created which could make the designers of dies more efficient in their design process. By taking advantage of the many design tools available today, designers could be in a working environment which does many of their mundane calculations and conversions for them, and which helps them to see their problems from different perspectives. This environment could also help them analyze their initial designs and provide feedback for them to improve their die designs before sending them to the tool and die shop.

One of these design aids was examined extensively in this thesis and it was suggested that it could be used as a method of early feedback for placement of gates, vents and overflows. This device, which uses the medial axis transform, the distance map, and related computer generated entities, should allow the designer to see the flow patterns, in a coarse sense, of the part being cast. Since the feedback from the analysis which uses this tool is very quick, (in the order of a few minutes) it could be a useful way to narrow down the possible choices for locations of the gates, overflows and vents.

Another potentially useful tool was also discussed, though less fully, in this thesis; use of the distance map as a visualization device for the designer to see which sections of the part to be cast are largest or thickest. The value of this tool should not be

underestimated, as the designer can use the visual information provided by it, in qualitative and quantitative ways, to improve the overall design process and design.

It should also be mentioned that both the distance transform and the medial axis transform are completely general. As long as the resolution of the grid points is sufficient, a distance transform and a medial axis transform can always be calculated for a valid geometric object. And since both can be calculated in $O(n)$ time, it seems they could possibly be used in other ways in many other applications, in both two and three dimensions.

6.2. Future Work

Including Mesh Thickness

Many die cast parts, such as housings or casings, have thin walls. The FEM code used for this thesis for potential flow in shell elements includes a capability to use a thickness variable for each element. The thickness of a thin-walled cast part at any medial axis point can be found using the distance transform value for that point; the thickness is simply twice the magnitude of the DT value.

Thus, on the flat mesh created for the simple flow analysis, one could include the section thickness at each node and make use of this capability of the FEM code. This would no doubt improve the accuracy of the analysis. Note that this use of thickness would be most suitable for thin-walled parts.

Automatic Mesh Generation

There are two potential kinds of automatic mesh generation which could use the medial axis transform. The first is the kind mentioned early, based on the central MAT

points, while the second is generation of a full finite element mesh based upon the boundary representation and its medial axis.

It was stated earlier that essentially the only part of the “simplified flow analysis” process which was not done by the computer but which ideally would have been, is generation of the flat mesh from the medial axis transform. This seems quite possible considering the nature of the medial axis, with its different central sections being differentiated by which external surfaces they are equidistant from. Successful work in this area would make this part of the user interface complete in a sense, and it seems likely it could have applications in other areas of engineering and computer science.

It would be useful if a full three dimensional mesh could be automatically generated from a model (computer or real) of the part. Ideally it could be made so that its format is the same as the format required by the full analysis part of the software. If the creation of the mesh is done at the same place the analysis software is used, an optimal mesh could be created for the analysis for which it will be used.

If the part exists as a solid model representation, its surface is defined completely. To get a surface representation of an actual part, the part could be scanned with a laser scanner. This surface representation is the most important part in creating a mesh.

If it is known how accurate the surface of the mesh to be used must be, the surface can be sectioned into the face elements of the mesh to be. A more difficult task is to create the inside mesh based on the surface mesh. This is where the medial axis transform could be of some use. The center MAT sections could be used as reference points on which to build the inside mesh.

This full three dimensional mesh generation seems a bit more difficult than the flat mesh generation mentioned earlier. It seems quite possible, however, and its usefulness would go far beyond the kinds of analysis done in a die casting design environment.

Orientation of the part

Orienting the part with respect to the die is important to remove unnecessary slides, where such slides exist. Determination of the minimum number of slides for a given object seems to be a problem of optimization and computational geometry. It might be worthwhile to the die casting industry to solve this problem in the general case, and to add it to the design environment described in this thesis.

Refining the Distance Map Grid

There are probably many cases of die cast parts which have fine detail in some areas of the part and little detail in others. It seems possible that such a part could be divided into sections based on such detail and each of these sections could use a grid with an appropriate resolution. This is done today in finite element analysis meshes to avoid making the entire mesh very fine, while allowing fine meshes in areas where there is significant detail, and keeping computational costs down. Many of the same ideas could be used for the distance map grid in order to speed up the calculations.

6.3. Contributions

The contributions made to the engineering research community in this thesis were:

1. Extension of the Medial Axis Transform to three dimensions,

2. Recognition and calculation of the central and boundary parts of the three dimensional Medial Axis Transform,
3. Use of the three dimensional Distance Map as a design and visualization tool in the die casting industry,
4. Use of the three dimensional Medial Axis Transform as a design and analysis tool in the die casting industry, and,
5. Use of a fast flow analysis tool for die casting die design.

References

- 1 Kaye, Alan and Street, Arthur, *Die Casting Metallurgy* - (Butterworths Monographs in Materials), Butterworths & Co. Ltd., 1982
- 2 Pokorny, H. H. and Thukkaram, P., *Gating Die Casting Dies*. 4th edition, Society of Die Casting Engineers Inc., 1981
- 3 Smith, Harold J., et al, *Metals Handbook*, 8th ed., volume 5, Forging and Casting, 1970, pp. 285-333
- 4 Sully, Lionel J. D., *Metals Handbook*, 9th ed., volume 15, Casting, 1988, pp. 286-295
- 5 Birch, J. M., 'High Pressure Die Casting of Non-ferrous Alloys', *Materials Science and Technology*, 4, 1988, pp. 218-221
- 6 EXCO Engineering, Discussions with Warren Bishenden, March 30, 1993.
- 7 Rawlins, Gregory J. E. and Wood, Derick, 'Ortho-Convexity and its Generalizations', from *Computational Morphology*, G. T. Toussaint, editor, Elsevier Science Publishers B. V., 1988
- 8 Danielsson, Per-Elik, 'Euclidean Distance Mapping', *Computer Graphics and Image Processing* 14, 1980, pp. 227-248
- 9 Borgefors, Gunilla, 'Distance Transformations in Arbitrary Dimensions', *Computer Vision, Graphics, and Image Processing*, 27, 1984, pp. 321-345
- 10 Sabersky, R. H., Acosta, J. A. and Hauptmann, E. G., *Fluid Flow - A First Course in Fluid Mechanics*, 2nd edition, Macmillan Publishing Co., Inc., 1971
- 11 Cook, R. D., Malkus, D. S., and Plesha, M. E., *Concepts and Applications of Finite Element Analysis*, 3rd edition, John Wiley & Sons, Inc., 1989, pp 486-488

-
- 12 Gu, Maoshi, *Computational Weld Analysis for Long Welds*, Ph.D. thesis, Carleton University, 1992

END

1 6 - 1 2 - 9 3

FIN