A HYBRID DESIGN SUPPORT SYSTEM FOR DIE CASTING SCHEME

LU HONG YUAN

Ph.D.

THE HONG KONG

POLYTECHNIC UNIVERSITY

2003

The Hong Kong Polytechnic University

A HYBRID DESIGN SUPPORT SYSTEM FOR DIE CASTING SCHEME

By

Lu Hong Yuan

A thesis submitted to

The Hong Kong Polytechnic University in accordance with the regulations for the Degree of Doctor of Philosophy

Department of Industrial and Systems Engineering

The Hong Kong Polytechnic University

March 2003

ABSTRACT of thesis entitled "A Hybrid Design Support System for Die Casting Scheme"

Submitted by Lu Hong Yuan

for the degree of Doctor of Philosophy

at The Hong Kong Polytechnic University in August, 2002.

The increasing demand for casting quality, rapid production, process complexity and low cost requires continuous improvement of the die casting process. However, the complexity of the casting process rules out that an optimal solution can be solved purely by an empirical and analytical approach. Development and application of a computerized system is an important step to automate the die design process and compensate for the lack of design experts. This thesis employs the knowledge and feature-based approach, fuzzy judgment technique and computer-aided method to present a hybrid design support system (HDSS) for a die casting scheme. The design of a die casting scheme is a critical phase in die design. Die casting schemes often determine the successfulness of a die casting die. The design of the die casting scheme includes three parts, i.e., parting scheme design, gating system configuration and selection of filling parameters. In this thesis, the development of a hybrid design support system is described.

The choice of a suitable parting scheme is a crucial step in determining the layout of a die and gating system. A parting scheme is mainly influenced by concave features in die casting. A concave feature may prevent the removal of a die casting part from a die cavity and influence the use of side cores. Geometrical feature analysis involves detection of concave features and determination of their release directions. A ray detection method is proposed for recognizing and extracting concave features from a die casting. With this ray detection method, each of the concave feature is assigned a block factor. A set of concave regions and their hulls can be built by examining the block factors. The points in a concave hull carry geometrical information required for the determination of normal vector and release direction. The release direction denotes a direction or a set of directions along which the concave region can be released. Each concave feature can correspond to a specific release direction or a set of release directions. A graphical method, release direction (RD) map, is used to study release direction. Such graphical methods can represent clear contours of the release directions. Based on the translation characteristic of a rigid object in mechanics, the release direction of die cast objects is deduced. The minimal release direction principle and vector plane approach are proposed to solve for the release direction of concave regions.

Parting scheme is a geometrical feature dependent in nature. In the present study, the parting scheme design is divided into three stages, parting scheme generation, parting scheme evaluation and parting scheme optimization. In the stage of parting scheme generation, the direction-shifting approach is used to generate the parting schemes. Each possible parting direction will match one or more planes as possible parting schemes based on heuristic knowledge or design rules. All possible parting schemes can be generated by shifting all of the possible parting directions.

The system allows designers to evaluate the performance of parting schemes. The performance of a parting scheme is identified by multiple design rules. In order to evaluate the parting schemes, a series of evaluation models is developed by formulating design rules as satisfaction or test functions. Then the parting scheme is evaluated by the function values, each of which reflects an aspect of the performance of a parting scheme. Thus a quantitative measurement on parting scheme performance can be realized. A multi-criteria weighted optimization model is used for the determination of an optimal parting scheme from candidate parting schemes. Analytical formulae are used to model the parting scheme of casting dies.

Before a gating system is designed, the value of the parameters should be selected. A fuzzy synthetic evaluation technique is proposed for the selection of process parameters. The fuzzy synthetic evaluation divides factors and value domain of parameter into discrete quantities to create a fuzzy factor set and an alternative set. By fuzzy transform, the contribution of each discrete level and each factor can be synthetically considered. The fuzzy synthetic evaluation can give reasonable and consistent result if the factor set and alternative set are provided. The proposed approach overcomes the shortcoming of the conventional method in the selection of die casting parameters.

In development of gating design system, a knowledge-based technique is applied. The case-based approach and feature-based CAD approach are combined to generate a gating configuration. The case-based approach is used to match the successful cases of gating configuration by case reasoning. With reference to the matched case, a new gating system can be designed. The feature-based CAD approach is used to design a gating system under an interactive CAD environment based on casting feature and gating knowledge. A knowledge representation scheme based on the knowledge object is developed for gating configuration. The hybrid knowledge representation can integrate the feature representation and inference in a knowledge object. The dimension dependency network is developed to define gating dimensions. A gating configuration is produced either by case-based approach or by feature-based CAD approach. The gating position, gating type and gating dimensions can be recommended. In the thesis, the proposed methodology was tested based on real castings from industry and the recommended designs were found to be capable of producing sound and acceptable castings.

The above exploratory work links the knowledge of different disciplines to develop a hybrid design support system for a die casting scheme. The hybrid approach proposed in this thesis is one of the first of its kind and contributes to the automation of CAD system in die casting die design.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor, Professor W.B. Lee, for his enthusiastic and unfailing support. His patient guidance, encouragement and far-reaching vision have made the often arduous task of graduate study worthwhile. Special thanks are also due to the Research Committee of the Hong Kong Polytechnic for the award of the scholarship.

I am particularly indebted to my organization, Shenyang Research Institute of Foundry, for the continuous support to my Ph.D. study.

I would like to express my gratitude to my colleagues and friends Dr. To Suet, Dr. M.J. Cai, Dr. X.Y. Wen, Dr. C.F. Cheung, Dr. J.G.. Li, Dr. D. Gao, Dr. S.M. Mei, Dr. J. Duan, Dr. S.Z. Wu, Dr. J.P. Fan, Ms. H.B. Tang and others inside and outside The Hong Kong Polytechnic University for their encouragement and support during the course of the work.

Finally I want to thank my parents, my wife, my daughter and all members of my family. Without their steady support, this thesis would not have been completed.

Table of Contents

Abstract.....i
Acknowledgement.....v
Table of contents.....vi
List of figures.....ix
Listof tables.....xii
Nomenclature.....xiii

Chapter 1 Intoduction.....1

1.1 Basic statement.....1
1.2 Objectives.....2
1.3 Research approach.....4
1.4 Organization.....5

Chapter 2 Background and Overview.....7
2.1 Description of a die casting scheme and its design.....7
2.2 Survey of related works.....9
2.2.1 Computer-aided gating design.....9
2.2.2 Computer-aided parting analysis.....12
2.2.3 Computer simulation of die casting process.....17
2.2.4 Problems in existing works.....19
2.3 Hybrid design support system for a die casting scheme.....21
2.3.1 Description of hybrid design support system.....21
2.3.2 Requirements for the hybrid design support system.....22
2.3.3 Framework of the hybrid design support system.....23

Chapter 3 Analysis on Geometric Feature and Release Direction.....28
3.1 Problem description.....28
3.2 Analysis of removal of a die casting.....30
3.3 RD of a particle point.....32
3.4 RD of a plane curve.....34
3.5 RD of a curved plane.....40
3.6 RD of a concave region.....49
3.7 Detection of concave regions.....51
3.7.1 Definition of concave region.....52
3.7.2 Detection of concave region.....53
3.7.3 Extraction of concave region.....56
3.8 RD distribution and parting direction.....58
3.8.1 RD distribution of a casting.....58
3.8.2 Influence of RD distribution on parting direction.....63

3.9 Summary....67

Chapter 4 Generation of a Parting Scheme....69
4.1 Basic description....69
4.2 Modelling of the die casting part....69
4.3 Fundamental features....71
4.3.1 Simple CR (SCR)....72
4.3.2 Complex CR (CCR)....74
4.3.3 Parting plane feature....77
4.4 Philosophy of parting scheme generation....80
4.5 Generation of parting schemes....81
4.5.1 Design variables and design space of a parting scheme....83
4.5.2 Generation mechanism of parting schemes....85
4.5.3 Activation of candidate parting directions....86
4.5.4 Selection of parting position....91
4.5.4.1 Parting position rule....91
4.5.4.2 Parting position modes....92
4.5.5 Matching process....94
4.5.6 Design process control....96
4.6 Design examples....97
4.7 Summary....108

Chapter 5 Evaluation and Optimization of a Parting Scheme....110
5.1 Brief description....110
5.2 Parting scheme design rules....110
5.3 Modelling of the design rules....113
5.3.1 Model of design criteria....114
5.3.2 Test function of design constraints....120
5.4 Evaluation and optimization of parting schemes....122
5.5 Case analysis....125
5.6 Evaluation and optimization process control....131
5.7 Summary....132

Chapter 6 Parametric design of process parameters....134
6.1 Basic description....134
6.2 Mechanism of fuzzy synthetic evaluation....136
6.3 Establishment of fuzzy sets....139
6.4 Establishment of alternative sets....144
6.5 Primary synthetic fuzzy evaluation....145
6.6 Secondary synthetic fuzzy evaluation....148
6.7 Application of fuzzy synthetic evaluation....151
6.8 Summary....153

Chapter 7 Configuration of a gating system..154
7.1 Basic description..154
7.2 Basic principle of feature-based gating CAD..155
7.2.1 Knowledge representation..156
7.2.2 Architecture of feature-based gating CAD..160
7.2.3 Mechanism of feature-based gating CAD..161
7.2.3.1 Determination of gating position..162
7.2.3.2 Selection of gating type..167
7.2.3.3 Determination of gating dimensions..169
7.3 Case-based gating design..183
7.3.1 Case library structure..183
7.3.2 Index and retrieval of cases..186
7.3.3 Case adaptation..187
7.4 Gating example with feature-based CAD approach..188
7.5 Verification and modification of the die casting scheme..193
7.5.1 Computer simulation..194
7.5.2 Modification of the die casting scheme..197
7.6 Summary..198

Chapter 8. Conclusions and further work..200
8.1 Overall discussion and contributions..200
8.2 Summary of research..203
8.3 Suggestion for further research..205

References..R1-R10

Appendices
Appendix 1 Algorithmic basis to solve for release direction..A1
Appendix 2 Some examples in case library..A7
Appendix 3 Program for construction of a die casting model..A10
Appendix 4 Program for calculation of filling parameters..A21

List of Figures

Fig. 2.1 DEVM model for die casting scheme design........23
Fig. 2.2 The framework of the modules for die casting scheme design........25
Fig. 3.1 Removal of a die casting with side cores........29
Fig. 3.2 Crs and RDs in a die casting........30
Fig. 3.3 RD of a point on cavity surface........33
Fig. 3.4 RD of a plane curve........39
Fig. 3.5 Examples of RD for different curves........40
Fig. 3.6 Discrete plane and RD of surface piece........43
Fig. 3.7 VP and VP cluster........45
Fig. 3.8 VP and RD map of a flat plane........47
Fig. 3.9 RD of a cylindrical sleeve........48
Fig. 3.10 RD of a spherical surface........48
Fig. 3.11 Demonstration of the solving procedure for RD_{CR} 51
Fig. 3.12 False count in declined plane case and outer surface of curved plane........54
Fig. 3.13 Ray detection method and typical points........55
Fig. 3.14 The CR with different CRI value........56
Fig. 3.15 CR recognition and extraction........57
Fig. 3.16 Four typical RD images........60
Fig. 3.17 Four relationships for any two images........61
Fig. 3.18 RD_{CR} distribution and RDDF value in different areas........63
Fig. 3.19 Directional interference Graph........65
Fig. 4.1 Primitives and casting........70
Fig. 4.2 The dependency graph of parting scheme design........81
Fig. 4.3 A principal candidate direction defined by target plane........89
Fig. 4.4 Parting position for four parting directions........93
Fig. 4.5 Parting modes in different features........94
Fig. 4.6 Matching procedure........95
Fig. 4.7 The control flow chart of candidate parting scheme generation........96
Fig. 4.8 A die casting and its CR shells........98
Fig. 4.9 The candidate parting directions and RD distribution........100
Fig. 4.10 Three pairs of *pcpd*s........102
Fig. 4.11 Surfaces with maximal projected area in $pcpd_1$........103
Fig. 4.12 Parting feature........103
Fig. 4.13 Parting scheme for $pcpd_1$........103
Fig. 4.14 Maximal projected area in $pcpd_2$ and $pcpd_3$........104
Fig. 4.15 Parting scheme for $pcpd_2$........105
Fig. 4.16 Parting scheme for $pcpd_3$........105
Fig. 4.17 Die casting model of a casing body........107
Fig. 4.18 Complex CR shells, RD and *pcpd*........107
Fig. 4.19 Two candidate parting schemes........108
Fig. 5.1 Example for calculating *Sat*(ps) of parting plane........117

Fig. 5.2 An example for evaluation of casting height..119
Fig. 5.3 The parameters for calculation of F_{core}..120
Fig. 5.4 Scheme A chosen as optimal parting scheme..128
Fig. 5.5 Scheme B chosen as optimal parting scheme..130
Fig. 5.6 Dimension of the casing body..130
Fig. 5.7 The control flow chart of parting scheme evaluation..131
Fig. 6.1 Filling velocity at different wall thickness and flow length..136
Fig. 6.2 Die casting parameter design with fuzzy synthetic evaluation..139
Fig. 6.3 The flow chart of the program of parameter design..152
Fig. 7.1 The framework of gating design..156
Fig. 7.2 Knowledge object..158
Fig. 7.3 Architecture of feature-based gating CAD..161
Fig. 7.4 Classification of parting primitives..162
Fig. 7.5 Recommended gating position for plate parting plane..163
Fig. 7.6 Recommended gating position for sleeve parting plane..164
Fig. 7.7 Recommended gating position for sphere crown..165
Fig. 7.8 Recommended gating position for U parting plane..165
Fig. 7.9 Gating position inference graph..166
Fig. 7.10 Four types of gating system..167
Fig. 7.11 Two different connections..168
Fig. 7.12 Gating type inference graph..169
Fig. 7.13 Selection of connection type..169
Fig. 7.14 A typical P-Q^2 diagram for a specific machine..172
Fig. 7.15 Determination of ingate size..175
Fig. 7.16 Tapered tangential runner..176
Fig. 7.17 The dimension dependency network in tapered tangential runner..177
Fig. 7.18 Relation functions used in the dimension dependency network
of tapered tangential runner..178
Fig. 7.19 Two different fan runners..180
Fig. 7.20 The dimension dependency network of fan gate..180
Fig. 7.21 Relation functions used in the dimension dependency network of fan gate..181
Fig. 7.22 The knowledge text for selection of R_{lfe}..182
Fig. 7.23 The case description of a tap body casting..184
Fig. 7.24 The solution of gating configuration for tap body..185
Fig. 7.25 Diagram of a basic class of plate parting feature..185
Fig. 7.26 The flow chart of case-based gating design..189
Fig. 7.27 Tapered tangential gate configuration..193
Fig. 7.28 Casing body produced with the given die casting scheme..193
Fig. 7.29 Filling patterns with different ingate positions..195
Fig. 7.30 Simulation result of filling process of casing body..196
Fig. A1.1 The system coordinate used in coordinate rotation transformation..A6
Fig. A2.1 A casing body of aluminum alloy with a fan gate..A7
Fig. A2.2 A notebook computer cover of magnesium alloy with a tapered tangential runner....A8
Fig. A2.3 A container casting of zinc alloy with a tapered tangential runner..A9
Fig, A3.1 An interface for construction of die casting model..A20

Fig. A4.1 An interface for casting characteristic input..A25
Fig. A4.2 An interface of calculation display...A25

List of Tables

Table 3.1 The rays hitting on the casting and CRI value of each point..55
Table 4.1 The RD corresponding to each SCR..73
Table 4.2 The RD corresponding to each CCR..75
Table 4.3 Parting position feature in different primitives..78
Table 4.4 The CR and *pcpd*..89
Table 4.5 Feature and RD of each CR..99
Table 4.6 RDDF value of each region..101
Table 5.1 *Sat*(ps) of four typical parting planes..117
Table 5.2 *Sat*(cast_h) for two typical schemes..119
Table 5.3 The first set of sizes of support frame die casting..125
Table 5.4 Volume of eight CRs..125
Table 5.5 *Sat*(num_sc) and *Sat*(vol_sc) of parting schemes..125
Table 5.6 *Sat*(ps), *Sat*(cast_h) and *Sat*(hold_for) of parting schemes..126
Table 5.7 The results of four test functions..126
Table 5.8 The weighted coefficients used in the optimization..127
Table 5.9 The second set of sizes of support frame die casting..128
Table 5.10 Volume of eight CRs..128
Table 5.11 *Sat*(num_sc) and *Sat*(vol_sc) of parting schemes..129
Table 5.12 *Sat*(ps), *Sat*(cast_h) and *Sat*(hold_for) of parting schemes..129
Table 5.13 Evaluation and optimization results of the casing body casting..131
Table 6.1 Selection of filling time..136
Table 6.2 Selection of filling velocity..136
Table 6.3 Factors and levels..141
Table 6.4 Grade of membership for critical values of wall thickness..144
Table 6.5 Grade of membership for critical values of casting size..143
Table 6.6 Grade of membership for critical numbers of casting entities..144
Table 6.7 Grade of membership of each level for three factors..148
Table 6.8 General format of input/output..151
Table 7.1 Characteristics of machine F580c..191
Table 7.2 Data from P-Q^2 analysis of machine 580c..191
Table 7.3 Required parameters..192
Table 7.4 Major dimensions of gating elements..192

Nomenclature

α	angle included between the normal and the positive direction of x-axis
φ	angle included between two extreme vectors
μ_{ij}	grade of the membership of factor in level
A_g	ingate area
A_i	level weighting set of factor *i*
a_{ij}	weighted value of level *j*, factor *i*
A_{pro_ps}	projection area of parting plane
A_{ps}	actual area of parting plane
B	matrix of primary fuzzy evaluation
B_i	primary fuzzy evaluation set of factor *i*
b_{ik}	element *k* in primary fuzzy evaluation set
CCR	complex concave region
cpd	a candidate parting direction
CPD	set of candidate parting directions
cpp	a candidate parting plane
cps	a candidate parting scheme
CPS	set of candidate parting schemes
CR	concave region
CRI	value of concave region indicator
D_g	gate thickness
D_{gmax}	allowable maximum gate depth

D_{gmin}	allowable minimum gate depth
h	casting thickness in gate area
h_{max}	maximal casting height between cover die and ejector die
H_{max}	average of two maximal sizes in x and y direction in parting plane
h_{max}	maximum casting thickness in gate area
h_{min}	minimum casting thickness in gate area
k	coefficient of the ingate thickness
L_c	a characteristic length in a casting
MRDP	point of minimal release direction
$\boldsymbol{n}_{e1}$	extreme vector 1
$\boldsymbol{n}_{e2}$	extreme vector 2
pcpd	a principal candidate parting direction
PCPD	set of principal candidate parting directions
RD	release direction
RDDF	distribution factor of release direction
RD_l	release direction of a curve
RD_P	release direction of a plane
RD_p	release direction of a point on a surface
RDS	release direction image on release direction map
R_i	level evaluation matrix for factor *i*
R_{ij}	level evaluation set for level *j*, factor *i*
r_{ijk}	grade of membership of level *j*, factor *i*, in element *k*
Sat(cast_h)	satisfactory function of casting height

Sat(dp)	test function of datum plane
Sat(ha)	test function of accurate area
Sat(hold_for)	satisfactory function of holding force
Sat(mp)	test function of machining plane
Sat(num_sc)	satisfactory function of core number
Sat(pm)	test function of parting mark
Sat(ps)	satisfactory function of parting plane
Sat(vol_sc)	satisfactory function of core volume
SCR	simple concave region
t	filling time
u_i	factor *i* in factor set
V	cavity volume
v_g	ingate velocity
v_k	value of the element k in alternative set
VP	vector plane
W_g	width of the ingate

For tapered tangential runner:

α(RBN)	angle of runner end
A(MR)	area of main runner
A(RBI)	inlet area of branch runner
A(RBN)	area of runner extension
d(DE)	exit thickness of delta region
d(MR)	depth of main runner

d(RBI) inlet depth of branch runner

d(RBN) depth of runner extension

d(RE) depth of runner extension

d(SA) diameter of shock absorber

dl(DE) inlet thickness of delta region

l(DE) length of delta region

l(RB) length of branch runner

l(RE) length of runner extension

r(RB) radius of branch runner

r(RBN) radius of runner end

R_{amr} cross-section area ratio between main runner and branch runner inlet area

R_{arbi} cross-section area ratio between branch runner inlet area and gate area

R_{arbn} cross-section area ratio between runner terminal and runner inlet

R_{dde} depth ratio between delta gate and ingate

R_{wde} width ratio between delta gate and ingate

R_{wrbn} ratio between the width and the depth of runner end

w(DE) width of delta region

w(MR) width of main runner

w(RBI) inlet width of branch runner

w(RBN) width of runner extension

w(RE) width of runner extension

wd(SA) depth of shock absorber

For fan gate:

A(FE)	area of fan gate
A(FI)	area of fan gate
d(FE)	depth of fan gate
d(FI)	depth of fan gate inlet
d(l)	depth of fan gate in specific height
L(FE)	length of fan gate
R_{afe}	cross-section area ratio between runner exit and inlet
R_{dfi}	ratio between width and depth of runner inlet
R_{lfe}	ratio between width of runner exit and overall length of runner
w(FE)	width of fan gate
w(FI)	width of fan gate inlet
w(l)	width of fan gate in specific height

CHAPTER 1

INTRODUCTION

1.1 Basic statement

In a mechanical design system, there are two different design stages: conceptual design and detail design. The detail design assigns the concrete shape and size to the components of a system, and the conceptual design includes the system planning, scheme determination, element layout and parameter selection *etc*, and often involves non-numerical or non-algorithmic knowledge. Comparatively, the conceptual design is more important and more difficult. If properly established, it can lead to superior system behavior. The quality of the conceptual design often dominates the performance of a mechanical system [Wang et al 1994a]. Similarly, the design of die casting die is also divided into two stages, the design of die in which components correspond to the detail design stage, and the design of die casting scheme which corresponds to the conceptual design stage. Conventionally, the design of a die casting scheme includes parting scheme design, gating system configuration and process parameter selection. Similarly, the die casting scheme often dominates the structure of a die and the success of a die casting production. Unfortunately, there is no analytical formula so far to model the design of a die casting scheme. Many phases of a die casting scheme design require empirical knowledge or skill of experts, and involve trial and error procedure [Barton 1981, Herman 1985, YiJi and Siauw 1997]. Errors in the design of a die casting scheme are difficult to be avoided.

An appropriate design of a die casting scheme is an essential pre-requisite for successful die casting production. If a die casting scheme is not properly designed, imperfect die castings will result and the die has to be rectified. This procedure is expensive and time consuming. In addition, there are increasing demands for shortening the development time, reducing development cost and providing quality castings, the efficient and error-free design of a die casting scheme is still a challenging work in a die casting process. The high demand for shorter design and manufacturing lead times, good dimensional and overall quality, and rapid design changes have become bottle-necks in the die casting industry [Fu et al 1999]. By automating the design process, the design lead times can be dramatically shortened. Development and application of design support system combining knowledge-based and computer-aided approach is an important step to automate and improve the design of die casting schemes. It is believed that such hybrid design support system can provide a useful tool for the die casting industry. All the above considerations motivate the present research.

1.2 Objectives

The design of a die casting scheme is a creative activity in die casting design. The support tool for the die casting scheme design is much more in need than it was in the past. The major purpose of the research is to develop a hybrid design support system to assist die casting scheme design. Since the system involves different knowledge and approaches, and covers different design phases of die casting scheme, the following specific tasks are aimed at:

i) To propose an approach for the recognition and extraction of geometrical features of die castings

The design of a die casting scheme is geometrical feature dependent. Geometrical features that influence a die casting scheme should be analyzed. A generic approach is proposed so as to recognize, extract and analyze the geometrical features required for die casting scheme design.

ii) To propose a parting scheme generation mechanism

A parting scheme design includes determination of parting direction and parting plane. In general, a parting scheme is not unique for a given casting. The proposed parting scheme generation mechanism should be able to match between feasible parting direction and parting plane so as to generate all possible parting schemes based on the geometrical feature and the design rules.

iii) To propose evaluation models of a parting scheme

After all candidate parting schemes are generated, an evaluation of them should be performed. Evaluation of a parting scheme involves comprehensive considerations of all design targets so as to identify an optimal or near optimal design scheme. For the judgment of a parting scheme, the performance of the parting scheme should be described quantitatively. Computable evaluation models are required to obtain quantitative evaluation information.

iv) To propose selection mechanism of process parameters

Since there are some drawbacks in selecting process parameters in conventional practice, a new approach, which is able to consider multiple factors and to work consistently, is proposed in the selection of process parameters. After the required characteristics of a die casting is known, the specific parameters can be addressed. Application of the new approach in selection of design parameters can provide more reasonable and economical results.

v) To develop a knowledge-based gating design

Since gating design is mainly based on empirical knowledge, a knowledge-based approach is required in the development of gating design system. Appropriate knowledge representation schemes and reasoning mechanism are proposed. Successful gating cases are collected and a case library is built.

By solving the issues above mentioned, this study would provide a strong support for development of a fully automatic system or virtual prototype system for the design of die casting dies. With this research, a complete solution of die casting scheme can be obtained.

1.3 Research approach

The design of die casting schemes heavily relies on experiential knowledge or expertise knowledge, it is difficult to make an error-free design. The conventional CAD approach is often not adequate. This study employs CAD technique, fuzzy theory, numerical simulation and knowledge-based approach to build a hybrid support environment for different phases of die casting scheme design. The CAD technique is used to construct a part model. The fuzzy synthetic evaluation algorithm outperforms conventional methods when it is appropriately applied to solve the problems of fuzzy uncertainty [Zheru et al 1996], and it is used to select process parameters in the present research. Feature-based approach is efficient in the analysis of concerned geometrical features [Lok et al 1995, Zhang et al 1989]. It is used to recognize and extract concave features of die casting. The knowledge-based approach which has been recognized to be most potential for solving domain specific problems requiring expertise and empirical knowledge [Hu 1989] is used to build parting scheme design and gating design system. In order to verify the validity and to increase the reliability of a casting scheme design, numerical simulation is introduced to provide feedback information for the modification of design.

1.4 Organization

The thesis is divided into eight chapters. Chapter 1 provides the basic statement, motivation and objectives of the study. Chapter 2 gives the related work and describes the framework of the hybrid design support system. In Chapter 3, the approach of recognizing concave region and deriving release direction is proposed. In Chapter 4, the methodology for the generation of candidate parting schemes is proposed, the parting scheme generation is demonstrated. Chapter 5 discusses the general design rules used in parting scheme design and the modelling of the rules. Evaluation and optimization of parting scheme are described. Chapter 6 provides the fuzzy synthetic evaluation for the selection of parameters. Fuzzy set, fuzzy transform and fuzzy evaluation are introduced. In Chapter 7, a combination of case-based approach and feature-based CAD approach is proposed for the gating design.

Design principles and design cases are described. The overall discussion, research summary and recommendations for further work are given in Chapter 8.

CHAPTER 2
BACKGROUND AND OVERVIEW

2.1 Description of a die casting scheme and its design

The design of a die casting die can be divided into two stages, the design of a die component and the design of a die casting scheme. Die component design is the detailed design which determines the shape and dimensions of die components. Die scheme design is a conceptual design which involves the determination of structure, layout and process parameters. The detailed design can be performed only after a die casting scheme design has been completed. Conventionally, the design of die casting scheme includes

i) parting scheme design,

ii) gating system configuration, and

iii) parameter selection.

The design of a die casting scheme is a highly knowledge-intensive and creative activity. There are two characteristics for the design of a die casting scheme:

i) Single input/multiple output characteristic

A die casting scheme design is characterized by "single input/multiple output" in nature, that is, for a given die casting part, there may exist several possible schemes to be generated. For such a design problem, there are two difficulties in the development of a design support system, i.e., specific design decision space and comprehensive evaluation of the quality of the design schemes [Wang et al 1994a].

ii) Design inconsistence

For a given die casting part, the design may be different among different designers. This indicates that parting scheme design is heavily dependent on the intuition and experience of individual designers. The design knowledge representation is not easy to capture.

The procedure of a die casting scheme design involves four basic phases. These are: analysis of geometrical features, design of parting scheme, configuration of gating system and selection of process parameters. At first, the geometrical feature of a given die casting part is analyzed. The design of a die casting scheme is heavily dependent upon the geometry of the casting part. In this step, a comprehensive understanding of the geometrical features of a casting is required. After the geometrical feature has been grasped, the parting schemes are conceived. In this stage, many factors should be considered, and several candidate parting schemes may be visualized. The empirical knowledge and design rules are important in visualizing parting schemes. Then the evaluation on each scheme will be carried out such that both the strong and the weak points of the scheme are recognized. Once an optimal scheme is determined, the gating configuration will be performed. During the configuration of a gating system, some filling parameters should be defined, such as filling time, filling time and so on. The parameter selection requires a consideration of the specifications of the casting part. It is dependent on the casting size, weight, thickness and quality requirement of the casting.

The basic requirements for a die casting scheme design are simple structure and favorable filling of the die cavity. Over the years, the expertise and experience of casting scheme design has well been summarized and formulated as rules by Editor [1981], Ravi and Srinivasan [1990], Hill et al [1991], Davey and Hinduja [1991], Guleyupoglu and Hill [1995]. These rules have served as the design guidelines used in die casting scheme design.

From the characteristics of the die casting scheme design mentioned above, the development of a computerized design system is necessary to assist the designer in the design of a die casting scheme. Die casting is an important manufacturing process. Larger numbers of non-ferrous castings are produced by die casting processes. However, injection mouldings are competing with die castings. To become more competitive, die casters have to increase the design reliability, reduce development time, decrease cost, increase productivity and improve quality. This situation encourages die casting processes to enhance the way that die castings are made. Die casting practice has shown that CAD/CAE technique is an efficient tool to meet increasing stringent requirements. Since die casting scheme design is a crucial task in die design, the development of a design support system to automate die casting scheme design will be much needed for the die casting industry.

2.2 Survey of related work

Application of CAD/CAE package in die casting can dramatically reduce the development time and improve a die design process. In the past several decades, much work has been done on the development of CAD/CAE package and methodology. In this section, the major CAD/CAE work is surveyed and the problem is discussed.

2.2.1 Computer-aided gating design

The gating system is one of the most important components in a die casting die, and gating design is a key task in die design. In the development of the CAD tool for die design, much work has focused on gating design. The gating system provides the passage for molten metal to flow from the chamber to the cavity and has great influence in controlling the cavity-filling pattern. Many foundation researches have better made in gating technique [Earle 1960, Plyatskii 1961, Draper and Perkert 1962, Louis and Draper 1966, Kaiser et al

1971, Lindsey and Wallace 1972, Benedek 1979, Yang and Yao 1980, Yoshiaki et al 1989, Douglas 1997]. These researches provide many design formulas or guidelines either based on theoretical derivation or based on experiments. In the early 1970s, The International Lead Zinc Research Organization sponsored research to reduce the cost of die cast zinc parts and research work was conducted to determine how the gating system should be designed to minimize the entrapment of gas in metal and the time required to fill the die. The research led to the development of a CAD tool for the runners, gates and overflow system for thin-walled zinc die casting dies [George et al 1981]. The basic idea of the CAD tool is to divide the casting into a series of segments and make sure each segment receives an adequate quantity of molten metal flowing into it. The tapered tangential runner, fan gate and overflow have become widely adopted.

Metlflow software was developed by Moldflow Australia Pty. Ltd.. The software permits computer-aided design of the flow system for aluminum, zinc, and brass castings [Andresen 1986, 1987]. The Metlflow technology is based on the use of tapered tangential runners. Trapezoidal cross sections are standardized into four basic elements, that is, the joint runner, bend, junction, and tapered runner. A runner system is comprised of four elements. P-Q^2 method was utilized to relate the metal pumping capability of the casting machine and shot sleeve with the flow requirement of the die cavity.

Another soft package for gating system design was developed by Rao et al [1989]. The package is based on the P-Q^2 algorithm. Using the algorithm, the molten metal flow rate through runner and gate can be predicted for a given set of machine conditions. The package made use of the casting and machine parameters for the design of the gating and runner system of die casting dies. A choice of tapered runner, fan feed gate, and main runner system is available in the package.

A series of programs known as the GATEWAY was developed in B.N.F. Metals Technology Centre which involves gating design, filling time, gate velocity, weight estimating, cost and porosity estimation procedures [Allsop and Kennedy 1983]. Another CAD package including typical gating systems was developed by Lee & Chan [1989]. User can specify a particular type of gating configuration, then the package gives relevant data according to the information entered by the designer. CASTFLOW [Yuji and Siauw 1997] can estimate the expected flow time, gate speed, metal temperature and the percentage solidification to each zone of fill. There are some other similar packages developed with interactive design of the feeding system [George et al 1987, Jin et al 1997].

With the development of computer technique, more calculation and graph works were incorporated into CAD/CAE system. A CAD/CAE system developed by Zhang [1997] and Yue et al [1997] includes the gating design and the simulation of thermal field and flow field to predict the defects that would occur during die casting. The optimization of gate and runner design was studied for die casting dies by a simulation annealing algorithm [Lin and Tai 1998, Tai and Lin 1998, 1999]. The CAD package for gating design based solid modelling was developed by Ji et al [1997]. The system provides better graphic function.

Although many efforts have been made in the development of CAD systems for gating design, and a large number of CAD packages have been developed in the world, their principles and methods employed are basically similar. These packages were based on the algorithm of mathematical formulae and serve as a calculation and a graphical tool. Tedious calculation work can then be avoided. In fact, gating design needs the support of a more efficient CAD system. Knowledge-based system, as a branch of artificial intelligence (AI), promises a powerful tool in solving complicated design problems that would require significant human expertise for their solution [Jan 1992], and will produce designs that are better, and that are "right first time" more frequently [Pillinger et al 1991].

In recent years, the application of the knowledge-based approach has been found in related fields. A prototype knowledge-based system for the design of rigging system of light alloy castings was developed by Hill et al [1990 and 1991]. The system is based on four sets of design rules to recommend rigging system for simple castings. Another expert system is used for permanent mould tilt-pour casting and investment casting process [Nyamekye et al 1994, Hill et al. 1993]. Kondic [1993], Nanda [1994] and Er et al. [1996] proposed an expert system to assist casting product designers to select the most suitable casting process. In plastic injection mould and forging die design, a knowledge-based approach has also been developed [Dennis 1989, David 1990, Pillinger et al 1991, Lok et al 1995].

2.2.2 Computer-aided parting analysis

In order to meet the demand of the die casting industry, the function of the computer-aided system has been extending in recent years. The works focus on computer-aided parting analysis. The selection of a parting scheme is an important decision during the design of the die casting dies. This has attracted the attention of a number of researchers in the last few years. The issue of computer-aided parting design has been summarized by Nainy et al [1997]. The ultimate aim of the research is to develop a systematic method to automate the parting scheme design. As discussed earlier, the parting scheme includes parting direction and parting plane. In most research work, parting analysis is based on the parting direction. So far, several approaches have been developed for the selection of parting direction.

i) Specified parting direction

In some of the research works [Guleyupoglu et al 1985 and 1994, Lee and Smith 1987], the parting direction is specified by the user. For a given casting part to be analyzed, the user should decide the orientation of the part and specify a particular direction as parting direction at first, then the parting analysis is performed in the specified parting direction. Fischer et al [1997] proposed a primitive-decomposing approach for the determination of parting plane placement. The casting part is broken into primitives. After a parting direction is specified, the projection of the primitives is made along the parting direction. A number of parting line segments from the decomposed primitives can be obtained on a projected plane. Connecting parting line segments on each primitive creates parting plane. In Fischer's work, the orientation and decomposition of the part is carried out by manual interaction.

ii) Principal axial directions

This method uses the principal axial directions (x, y and z direction) as parting directions, and these directions are respectively alternated to analyze parting feasibility. With this method, a package involving parting line analysis was developed by Sirilertworakul et al [1993a]. The package analyzes the parting features by cutting the pattern into slices at intervals along each of the three axes. Undercut is distinguished by comparing the cross-sectional areas of successive sections. If the areas increase, decrease and then increase again, an undercut exists in the region of the small areas. This method may require a large number of sampling directions before an optimal parting direction is found since release directions for each undercut are not solved.

The orientation of a casting part or a pattern in a mould influences the position of the parting plane. In the development of a computer-aided system for the pattern design, the orientation of the pattern is analyzed in three principal directions [Guleyupoglu and Hill 1995 and 1997]. The orientation knowledge of a casting is modelled as four rules and is presented as four sets of fitness functions. The four fitness functions are calculated in the six primary directions respectively. According to the values calculated from the fitness function, a parting direction is suggested. Obviously, the parting direction is limited to the three principal directions.

iii) Normal directions of casting faces

The normal directions of faces on a part are considered as the feasible parting directions. In this method, the outward vectors of all faces are required for the selection of the best parting alternative. Hui and Tan [1992] explores the application of sweep operations to the mould design. By the use of solid-sweep operation, the mould core and cavity geometry can be obtained without the creation of undercuts or detached solid objects. A blocking factor and an undercut preference value are used to select a parting direction. In the algorithm, the possible parting directions are bound to one of the special directions such as face normal vectors or principal axes.

iv) Directions defined by visibility

Chen et al [1992, 1993] and Woo [1994] propose a visibility approach to analyze parting directions. The allowable parting directions are described by a Gaussian map (GMap) and its associated visibility map (VMap). The approach divides a given object into pockets for which the visibility and demouldability can be determined independently. The unit outward normal vector of each surface in a pocket creates a set of points on GMap. These points map to hemispheres on the Vmap that corresponds to the allowable parting directions of the surface. A pair of opposite directions that minimizes the number of cores is considered as optimal parting direction. This approach is demonstrated in plastic injection moulded parts [Winstein and Manoochehri 1996]. The part is divided into concave and convex regions to identify the allowable draw directions and parting line location. The effect of part geometry on parting line locations and draw direction is presented. In another approach, a pre-determined number of equi-distant nodes on the unit sphere are considered as draw direction alternatives, and the best alternative is selected by an energy optimization program [Ganter and Tuss 1990].

In the selection of a parting line, a basic approach is to locate the parting line at the outermost points or sections forming maximum periphery boundary when viewed along the parting direction. In order to identify such an outermost boundary, two methods are used in most research work, that is, orthogonal projection and vector judgment.

i) Orthogonal projection

This method is based on the design rule that the parting line should coincide with the projected boundary of the component when viewed along the parting direction. In this method, the casting object is projected onto a plane perpendicular to the parting direction. The outermost boundary of the projection is determined and then is projected back to the object. The edges forming the outermost boundary on the object can then be determined. These edges are connected to form the parting line [Ravi and Srinivasan 1990, Fischer 1997].

ii) Vector judgment

A casting part can be divided into sub-regions of surfaces that are formed by either the fixed die half or by the moving die half. The regions formed by any one of two halves can be defined as the set of surfaces whose outward normals form a positive dot product with a particular parting direction. The regions formed by the other half can be defined as the set of surfaces whose outward normals form a negative dot product with the particular parting direction. Parting line is located at the intersection edge of the two types of surfaces [Weinstein and Manoochehri 1996]. This method can be described in another way. The parting line lies at the border between the visible and the invisible portion of a part when viewed from a particular parting direction [Tan et al 1990].

There are some other typical work on parting design. Based on a set of design rules, Ganter and Tuss [1990] proposed three types of searches for parting planes, i.e., the centre of gravity search, the principal axis search, and single direction search in a computer-assisted parting line developed for cast pattern production. This approach will fail if the object is not removable from the mould along one of the prescribed directions. Wuerger and Gadh [1997] use a similar approach to select part orientation and die-open direction by detecting features that obstruct possible opening directions and computing their restrictions with respect to the whole part. The application is limited to planar surfaces at present. By classifying the undercuts in an injection mould, the relationship between the undercut feature and their mouldability are described [Fu et al 1999]. Virtual prototyping approach has been found in analyzing geometrical mouldability and parting feature for near-net-shape manufactured parts by Yin et al [2001]. Vellapillil [1985] and Kotschi [1989] presented the factors relating to the parting line and corresponding algorithms in their respective studies. Ravi and Srinivasan [1990] presented nine influencing criteria for decision making in the selection of parting surface. Assessment of parting-surface alternatives using proposed criteria enables the designer to make a rational choice.

The parting scheme design is important because it affects all the subsequent steps in die design. Comparatively, the development of parting design system has received little attention in die casting. Further work is needed to extend and improve the parting design system.

2.2.3 Computer simulation of a die casting process

With the development of high speed computers and improvement of numerical techniques, enormous researches on numerical simulation of die casting processes have been carried out worldwide. The simulations are used to visualize various aspects of the process such as cavity filling, temperature distribution, metal solidification and microstructure, so as to predict the potential defects or the performance of castings. A series of the computational fluid dynamics techniques, such as SIMPLE SIMPLER, (Semi-Implicit Method for Pressure-Linked Eqations) [Patankar 1980], MAC (Mark And Cell) [Harlow et al 66], SMAC (Simplified Mark And Cell) [Anhtory et al 1970], and SOLA-VOF (Solution Algorithm-Volume of Fluid) [Hirt et al 1970 and 1981, Nichols et al 1980] were used for the development of simulation programs.

The effort of simulation was first made in a sand foundry. Hwang and Stoehr [1985 and 1987], Stoehr and Hwang [1988] applied SOLA-VOF technique to the castings in the shape of thin-wall plate and three-spoke wheels. The simulation of three-dimensional flow coupled with heat transfer for casting processes has been developed by many researchers [Upadhya and Paul 1992, Nagasaka et al 1989, Kallien et al 1992].

In pressure die casting, there is also a world-wide research interest in the simulation of metal flow and heat transfer in die casting dies. Anzai et al [1988], Kappel and Smalloum [1989], Yukio et al [1989], and Mitsushi et al [1991] used a direct finite difference method to analyze metal flow and to predict defects in a plate cavity. Koichi and Toshio [1990] employed an extended SMAC method to simulate almost three-dimensional transient flow in a die cavity. Front Marker Tracing (FMT) was proposed to predict cold shut defects. A numerical analysis covering flow, solidification and back air-pressure was developed based on a method of viscosity-dominated flow [Ohtsuka and Ono 1990]. A three dimensional simulation model was realized to analyze cavity filling processes using the Navier-Stokes equation and SMAC method. Simulation results are given for three types of cavities to predict the flow behaviour, air entrapment and vortex generation [Hiroyuki et al 1991]. A network technique involves the modification of the Navier-Stokes equations to present the metal flow analysis in the filling system of die casting [Sulaiman and Gethin 1992, Sulaiman and Tham 1997]. Flow visualization technique was often employed to verify computer simulation [Lu and Lee 1993, Itamura et al 1996]. The flow pattern of molten metal in the shot sleeve for cold chamber die casting machines was simulated with a variable mesh method [Kuo and Hwang 1998]. Models based on finite difference method and finite element method were proposed to simulate 3-dimensional temperature fields of die castings and deal with complicated water cooling and/or oil heating lines [Zhang et al 1996, Xu et al 1997]. Using FLOW-3D, a three dimensional simulation of fluid flow and heat transfer during the filling of die cavity was conducted. The cavity geometry and process conditions were selected for die wear experimental studies [Venkatesan and Shivpuri 1993]. Lothar and Mark [Lothar 1993] presented the results of fluid flow and heat transfer analysis using 3-D finite difference software package to calculate heat losses of the metal as it passes through the gating system and the cavity, solidification and defect areas in casting. A runner optimization design of a die casting die was proposed to simulate the temperature distribution [Lin and Tai 1998]. Similar researches can be found in many publications [Shivpuri 1991, Frayce et al 1993, Barone and Caulk 1993, Jong et al 1993, Chen et al 1994].

The simulation software tools can give an insight for metal flow in the cavity and cannot undertake the design task of a die casting scheme. Before a simulation is made, a casting scheme must be given. The major purpose of the computer simulation is to verify the casting scheme by predicting the flow behaviour, solidification pattern, casting defect and so on.

2.2.4 Problems in existing work

Although researchers world-wide have made progress on the development of computer-aided systems and analytical methodology, there are some problems or limitations in existing work.

i) The development of the gating design system focuses on the algorithm program under the assumption that a user knows enough about gating knowledge and is competent to design a gating system. The use of the packages developed is limited to professionals who are knowledgeable in die design. The major contributions of such packages are the computing and graphical functions. There is a lack of the support of the design knowledge and recommendation for the gating configuration.

ii) The parameter selection is usually based on a certain single factor. In fact, the determination of a parameter may be influenced by many other factors. The parameter selection based on a single factor loses the influence of other related factors. This may causes an unreasonable result.

iii) The selection of parting direction is important in the development of the gating design system. In some research work, the parting direction is either specified by the user or restricted to be along one of the three principal axes. The former approach requires the visualization skill and experience of the user. Besides, this approach may require a large number of sampling directions before a feasible direction is found. The drawback of the later approach is the impossibility of knowing whether a feasible direction or an optimal direction exists at all. In fact, this method falls into the manual selection of parting direction. The orientation of the casting part is a crucial factor in the later approach. Once the orientation of a part has been determined, the choice of the parting direction is limited to the three principal axial directions. This implies that some other feasible parting directions, which are inclined with respect to the principal axis, are excluded from the potential parting directions. In this case, the optimal may be omitted. The later approach fails whenever the object cannot be released from the mould along one of three principal axes.

iv) There are two approaches to the generation of a parting plane, i.e., projection approach and vector judgment approach. If the outermost projection boundary is formed by a vertical plane, or there is a vertical plane between two surfaces with positive and negative dot product with the particular parting direction respectively, the two approaches cannot give a specific solution.

v) In the visibility approach, the optimal parting direction is considered to be the direction with minimal number of side cores and the other factors influencing parting scheme determination are ignored. Some good solutions may be overlooked.

vi) An efficient method has not been proposed for dealing with the feature of curved planes for the analysis of parting direction. At present, all previous work on the parting analysis are limited to the planar surface feature or due to the complicity of the analysis for curved plane. Since castings including planar surfaces are very few, the limitation is vital in its application.

2.3 Hybrid design support system for a die casting scheme

2.3.1 Description of hybrid design support system

Virtual prototyping is an emerging technique which refers to the visualization of a product without actually making a physical prototype of the product. The product design and evaluation can be carried out in a virtual reality environment. Therefore low cost and short product development cycle can be obtained with better alternative design to meet stringent design requirements. It is attracting the attention of researchers in the casting field [Weibender 1997, Gary 1998, Zhou et al 2001]. The hybrid design support system (HDSS) can be regarded as complementary to virtual prototyping but avoids expensive virtual reality tools and devices. The design of a die casting scheme is a complex process involving many decision factors and heavily depends on the experience of the die designers. The building of a physical die and the conduction of tests are time-consuming and expensive. The HDDS employs CAD technique, fuzzy theory, numerical simulation, feature-based and knowledge-based approach and provides a comprehensive design environment which includes sufficient design knowledge to support the design activities of a die casting scheme. Since the system integrates different knowledge and adopts multiple approaches, and covers all phases of die casting scheme development, it is named as the hybrid design support system. The scheme alternatives can be generated, evaluated and modified in the system. The designer can consider different scheme alternatives throughout the design process and make a better design decision. Combining the simulation technique with the knowledge and feature-based approach, the HDSS can apply a numerical model of a die casting scheme instead of a real die object to analyze and verify the performance and the mouldability of different schemes in the early stages of development. The HDSS can reduce the development time of a die casting scheme, increase the design reliability, and avoid excessive trial and error work in real operation.

2.3.2 Requirements for the hybrid design support system

The design of a die casting scheme is of particular importance to ensure reliability of die performance. It involves different design phases and many decision factors. Although computer-aided tools exists for different phases of die casting scheme design, most of them are still dependent on the judgment of the user. The major contribution of most software packages is to reduce the tedious calculation and graphical generation and but not to address the essential design features of a die casting scheme. The separate development of the design packages render the integration to be difficult. The lack of communication between different design phases may result in an inconsistent solution. In addition, the CAD packages are still dependent heavily on the experience of the individual designer. Most of the available simulation software tools are designed to simulate casting processes with the assumption that the casting scheme has already been defined. Previous work in this area addresses individual design function, rather than to offer a complete solution for a die casting scheme. In fact, an ideal computerized system for the casting industry would be one that can produce the solid model, produce the casting system, select the process parameters, and verify the validity of the design [Guleyupoglu and Hill 1995]. A hybrid support system that integrates different design approaches offers a possible solution of the problem.

2.3.3 Framework of the hybrid design support system

The overall process of scheme design in the hybrid design support system can be summarized as four basic stages: design, evaluation, verification and modification, as illustrated in Fig. 2.1. The system works in a Design- Evaluation- Verification-Modification (DEVM) pattern.

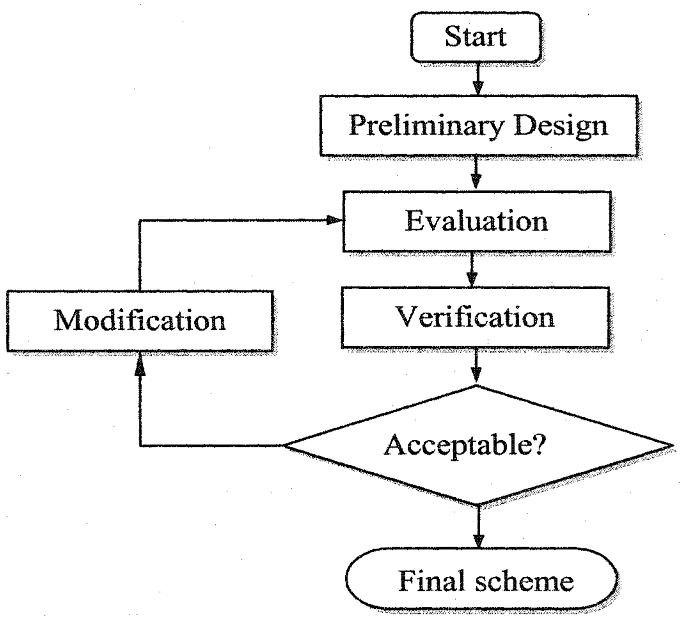


Fig.2.1 DEVM model for a die casting scheme design

<u>*Preliminary design*</u>: In the preliminary design stage, the scheme alternatives will be generated based on the geometrical feature analysis and specific design rules. The parting scheme optimization, gating configuration and parameter selection are involved in this stage.

<u>*Evaluation*</u>: Evaluation involves comprehensive consideration of all the design targets to

identify an optimal or near optimal design scheme among the possible scheme alternatives generated from the preliminary design. In this stage, quantitative analysis will be performed based on the evaluation models

Verification: The objective of verification is to check if there are any design errors in a given die casting scheme. This procedure requires trial production in real die casting operation or simulation of a die casting process. A simulation work is incorporated in this stage.

Modification: Modification uses feedback information from verification to analyze deviation and to modify the casting scheme. This procedure may repeat many times so that all deviations of the design are removed.

Fig. 2.2 shows an overview of the hybrid design support system illustrating the relationship among different components. The overall system consists of two main function blocks. The first block is a scheme design block, including four modules to perform different design tasks. The second block is an analysis block which consists of evaluation and analysis modules. There is also an user interface allowing communication between the die designer and the computer modules.

The functions of each module can be described as follows:

i) Module of geometrical feature analysis

The die casting scheme of a part depends on the geometry and is affected by the number and type of concave features. The purpose of the geometrical feature analysis module is to recognize and analyze the necessary geometrical features that are related to the scheme design, such as concave regions, release directions and characteristic plane.

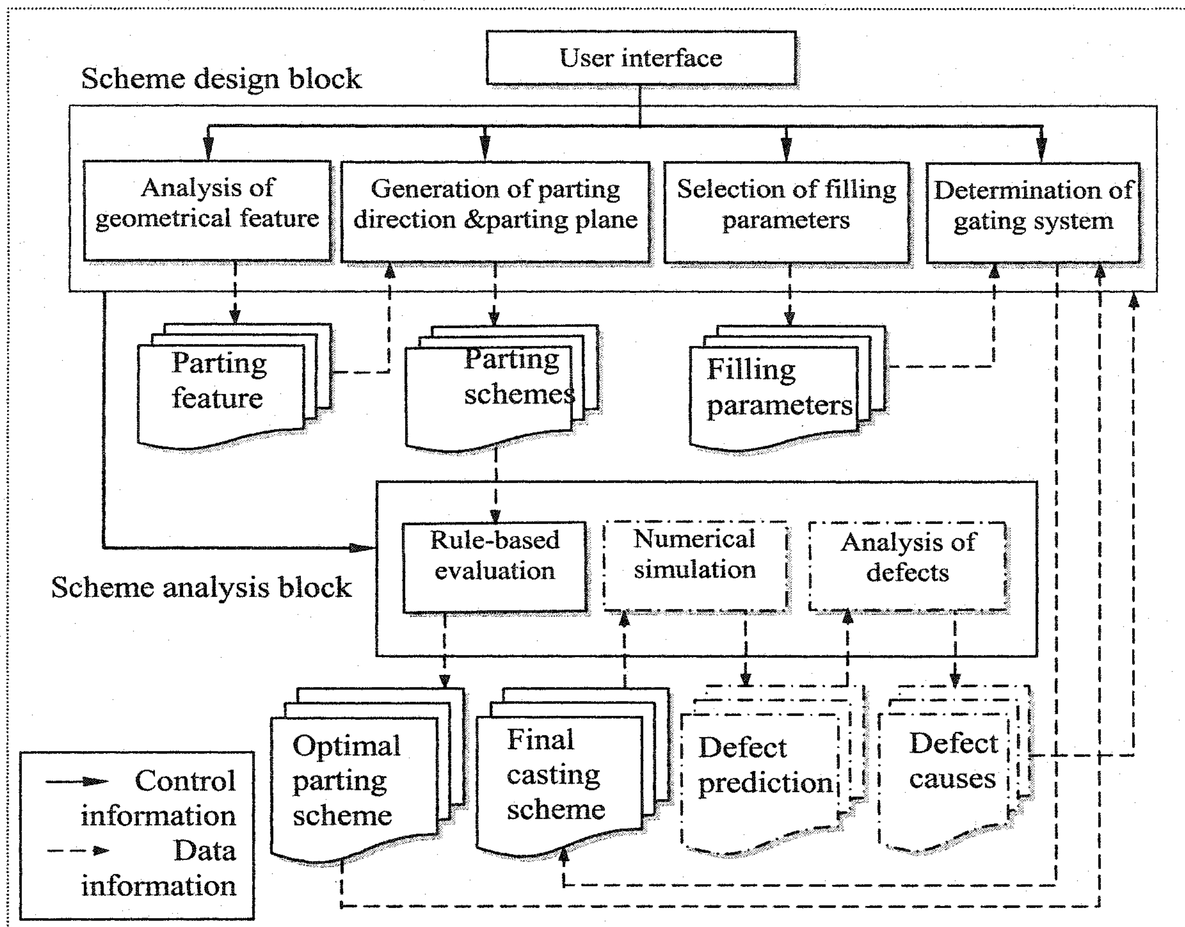


Fig. 2.2 Overview of the hybrid design support system

ii) Module of parting scheme design

The design of parting scheme includes the determination of the parting scheme and the parting plane. The module of parting scheme design uses the geometrical information from the module of geometrical feature analysis to generate all possible schemes based on the specific design rules.

iii) Module of filling parameter selection

This module is used to select process parameters which are useful in the design of the gating system and simulation procedure.

iv) Module of gating system configuration

The function of this module is to configure the gating system. The type, location and dimensions of the gating system can be suggested in this module.

v) Module of scheme evaluation

The function of this module is to perform a comprehensive evaluation on the generated design schemes. The evaluation is conducted based on the rule-based models and the performances of different schemes are judged. An optimal or near optimal scheme is proposed.

vi) Module of numerical simulation

Since a die casting process is invisible, and the outcome of a scheme can not be foreseen exactly, numerical simulation is used for this purpose. The deviations of the design will be tackled in this phase. Except for real trial production, numerical simulation is considered as a most effective approach for visualizing the filing process and verifying the validity of the die casting scheme.

vii) Module of deviation analysis

Based on the results from the simulation module, the errors of scheme design can be analyzed, and suggestions to be made improve the weak parts of the current scheme.

The last two modules in the analysis block serve as external programs, they are given by dot dashed boxes. The implementation of the system functions depends on the following critical factors:

i) recognition and extraction of geometrical feature
ii) calculation of release direction of concave regions

iii) mechanism of geometrical reasoning for die casting scheme design

iv) establishment of models of comprehensive evaluation

v) mechanism of gating design

CHAPTER 3
ANALYSIS OF GEOMETRICAL FEATURES AND RELEASE DIRECTION

3.1 Problem description

In die casting, molten metal is injected into a die cavity and solidifies in it. A die must be split into two main sections, i.e., ejector die and cover die, so that a solidified part can be removed out after the die opens. The part can be directly removed from the cavity after the die opens only if there is no interference between the die and the casting. The interference can be caused by a hole, slot, hollow and undercut occurring in the lateral wall of the die casting. Hole, slot, hollow and undercut are called concave regions (CR). The concave regions are formed by protruded components in the die, and therefore, result in the interference of relative motion between the die cavity and the casting part, which prevents the part removing from the cavity. To remove the casting, side slide cores are required. The side cores must be withdrawn from the die casting part before the part is removed, as shown in Fig. 3.1. Since the use of the side core increases the manufacturing and the maintenance cost, complicates the operation of cycles, and slows down the process, it is desirable to reduce the number of side cores whenever possible. Interference of motion occurs in the concave region, each concave region in a die casting can be regarded as a potential interference source. Whether a CR is a real interference depends on the direction along which the casting is released.

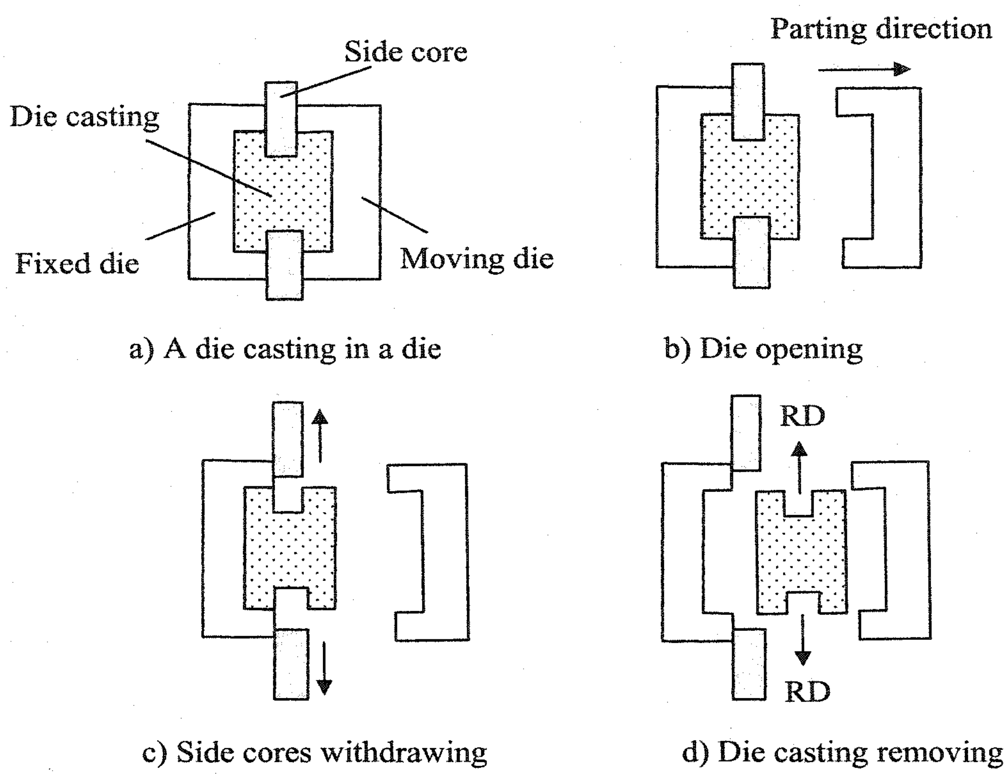


Fig.3.1 Removal of a die casting with side cores

For each CR, there corresponds a direction or a set of directions to release the core [Lu and Lee 2000a and 2000b]. This direction is called the release direction (RD) of the CR. The CR and RD distribution influence the determination of the parting scheme of die casting die, both of them must be identified before the parting scheme, including parting direction and parting plane, is designed. Type and number of CR and corresponding RD are dependent on the geometrical feature of a die casting. Fig. 3.2 illustrates the CRs and the RDs in a die casting. There are four CRs in the casting, CR_1~CR_4, corresponding RDs are RD_1~RD_4 respectively. The dotted lines denote the surfaces creating the CR. In x-z plane, CR_1 and CR_3 present a set of RDs, within ***a*** and ***b***, ***c*** and ***d*** respectively. CR_2 and CR_4 present an unique RD, RD_2 and RD_4 respectively.

Essentially, the design of parting scheme can be described as the problem to find a direction from RDs as parting direction which is optimal to the die casting process. Therefore, recognition of CRs and determination of RDs are essential steps for automatic parting scheme design. In this chapter, a general approach for the recognition of CR and determination of RD will be developed.

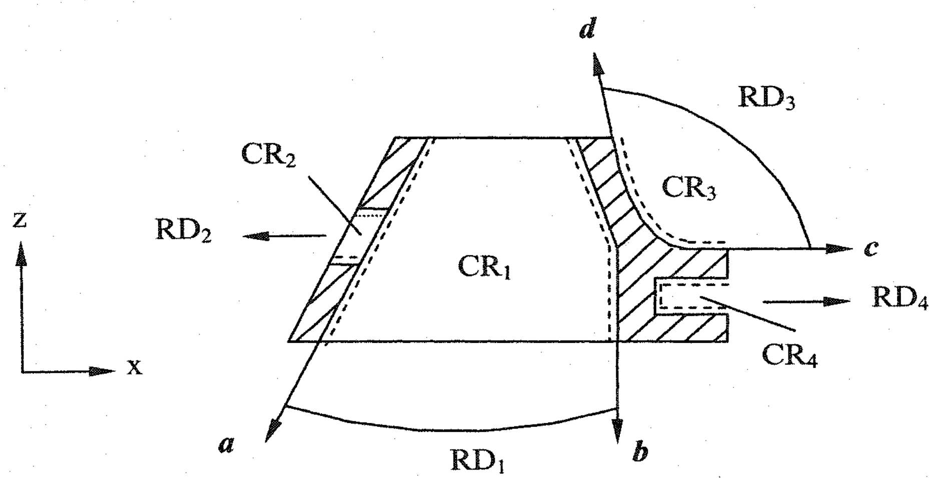


Fig.3.2 CRs and RDs in a die casting

3.2 Analysis of removal of a die casting

In general, the removal of a die casting part from a die cavity is a rectilinear motion, a special case of translation motion. When a die casting is removed from a cavity, the die casting has solidified completely. The solidified die casting can be considered as a rigid body. In mechanics, the translation of a rigid body is defined as follows: A rigid body moves from an initial position to a final one without any change in the orientation. During movement, every point on the body has identical magnitude and direction of displacement. All points on the body move in the same way [Kiran 1988]. In other words, motion of any point in the body can represents that of the entire body. For the translation of a rigid body, the body is often regarded as a rigid particle point. Any constraint exerted on the body is equivalent to exerting the constraint on the point representing the body. In die casting, the casting part is a rigid object and the removal of the casting is through a rectilinear motion. Therefore, the removal of a casting from the die cavity should follow the characteristic of the translation motion of a rigid body. Therefore, the following deduction can be made:

A die casting can be removed from a die cavity in a direction or a set of directions, if and only if all points on the die casting can be removed in the direction or the set of directions.

The removal of a die casting from a cavity is to release the casting from a die cavity. Interference between the die and the casting during removal occurs on the surfaces of the casting or die cavity. For this reason, the points, hereinabove and hereinafter, refer to those on the die casting or cavity surface. The RD of a CR depends on the orientation and geometry of the CRs, to be exact, on the points on the CR surfaces. CR surface dominates RD. RD can be calculated by examining the feature of the CR surfaces.

In order to derive the RD of a CR or a casting, the analysis of RD in different objects will be discussed below with a RD intersecting map, called the RD map. The RD map, which is a graphical method, is used to illustrate the solution of RD. Construction of RD map includes the following steps:

i) For a given CR, define the RD of relative points on the CR surface;

ii) Draw the RD at each relative point;

iii) Translate RDs of all relative points with their starting points coinciding with the origin of the Cartesian coordinate system,

iv) Map each RD on to unit sphere;

v) Examine the image created on the sphere surface, which represents the RD and is called the RD map.

The release direction (RD) of a CR depends on the RDs of the relative points on the CR surfaces, the analysis of the RD problem commences with point.

3.3 RD of a particle point

Let p be a rigid particle point in space. In the view of mechanics, it has six degree of freedom. If rotation is not allowed for the p, only three freedoms remain and are noted as ***X, Y,*** and ***Z*** respectively [Wang 1994a]. Then, the allowable motion directions of the point are all radial directions from the point to infinity. Let p be a particle point of a casting surface, and lie on the cavity surface. The cavity surface is rigid and smooth, it exerts a constraint on p. No motion is allowed towards the direction of inward normal of the cavity surface for the point p. Let an unit sphere be centred at the p, and rays be emanated in radial directions from the centre, the rays represent the allowable movement directions, i.e., release direction (RD) of the point *p*. The RD rays intersect the unit sphere and can be expressed as the spherical surface area [Kai 1990] or a solid angle subtended by the spherical surface [Ngoi 1997]. Obviously, the RD of the *p* on a casting surface corresponds to a hemispherical surface, 2 π steradians, as shown in Fig. 3.3.

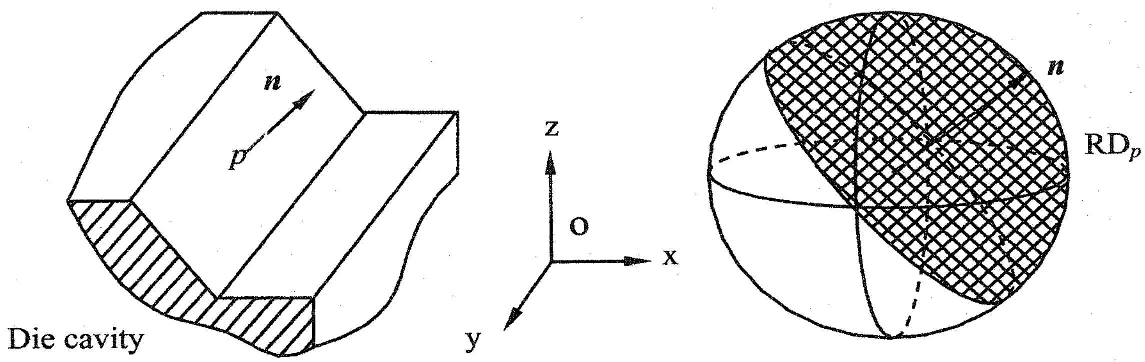


Fig. 3.3 RD of a point on cavity surface

Let $\boldsymbol{n}$ denote the inward normal vector of a casting surface or outward normal of cavity surface at a point *p*, then the RD of the point *p* on the surface can be defined as follows: The set of all directions with angular deviation of π/2 from the inward normal vector $\boldsymbol{n}$ of the casting surface at point *p* is the RD of the point, noted as RD$_p$, i.e.,

$$\boldsymbol{RD}_p = \{ \boldsymbol{d} : | \boldsymbol{n} \wedge \boldsymbol{d} | \leq \pi / 2 \} \tag{3-1}$$

where $\boldsymbol{d}$ denotes direction vectors at point in which the point can move.

The manifestation of the RD shown in Fig. 3.3 is called the RD map of a point. The RD*p* map of a point on a casting surface is a hemispherical surface. RDp is a set of directions.

3.4 RD of a plane curve

Suppose that L={p_i : i =1, 2, 3, ...} represents a plane curve (2-D curved line) extracted from a casting surface, and p_i be a point on the curve. The removal of the curve from a die cavity is also a translation motion. The characteristics of translation of a rigid body remain for the removal of the curve. Apparently, a rigid curve can be removed from a die cavity in a direction or a set of directions, if and only if all the points on the curve can be removed in the direction or the set of directions. Therefore, the RD of a curve, RD$_L$, equals the intersection of all RDps on the curve, i.e.,

$$RD_L = \{RD_{p1} \cap RD_{p2} \cap \ldots \cap RD_{pi} \ldots\} \tag{3-2}$$

where RDp is the RD of each point on the curve respectively. RD$_L$ can be determined by examining the RDp of all the points on the curve and finding their intersection. As defined previously, the RD of a point on a casting surface is a hemispherical surface on RD map, therefore RDp_1 , RDp_2 , ..., RDp_n are known and given by Equation (3-1) provided each normal vector $\boldsymbol{n}$ is determined. Thus their intersection can be solved once the normal vector $\boldsymbol{n}$ at each point p_i is obtained.

In fact, a curve includes infinite points, it is impossible to solve the intersection for RD$_p s$ of infinite points. In the set theory, the intersection of a number of sets means the shared elements by the sets [Group 1979]. Therefore, it can be solved by finding two sets with minimum shared elements if the two sets exist. By intersecting the two sets, resulting elements are the intersection of all the sets. A RDp is really a set of directions. Similarly, if such two sets can be found, which contain directions shared by all RDps, only the two direction sets are needed to determine the intersection of all points on the curve, the problem becomes simple. In fact, such sets do exist. In order to locate such sets, the RD problem for the case of a plane curve is considered. Suppose an arbitrary plane curve is defined by a single variable function

$$y = f(x) \tag{3-3}$$

and the function is continuous and monotropic in given domain. From Equation (3-1) and (3-2), the magnitude of intersection of RDps for all points depends on the orientation of the normal $\boldsymbol{n}$ of each points on the curvilinear. Obviously, the larger the difference between the orientations of each normal $\boldsymbol{n}$ is, the smaller the magnitude of the intersection of RDps will be. If the difference of the orientations is represented as the angle included between the normal directions, the two normal vectors with the largest included angle will present minimal shared RD. Therefore, the RD of the curvilinear, RD$_L$, can be determined by the intersection of RDs defined by the two normal vectors with the largest included angle. The two direction sets are the sets required. All the points on the curve can be released along the intersection of the two direction sets, i.e., RD$_L$. The above analysis leads to a minimal RD principle and extreme vector conception in the determination of RD for a curvilinear:

For a given plane curve, there exist at least two points on the curve, the intersection of RD defined by vectors at the two points keeps the minimal. Such points are called minimum RD points, referred to as $MRDP_1$ and $MRDP_2$, the vectors at the two MRDP points are called extreme vectors, referred to as $\boldsymbol{n}_{e1}$ and $\boldsymbol{n}_{e2}$, as shown in Fig. 3.4. The RD of the curve, RD_L, is determined only by the intersection of the two RDs defined by extreme vectors. In general, there must exist at least a pair of extreme vectors between which the included angle is maximal or the intersection of RD is minimal unless the case under consideration is a straight line. Therefore, RD_L of a plane curve can be expressed as follows:

$$RD_L = \{RD_{ne1} \cap RD_{ne2}\} \tag{3-4}$$

where RD_{ne1} and RD_{ne2} are the RD defined by two extreme vectors at point $MRDP_1$ and $MRDP_2$ respectively. It should be noted that RD_{ne1} and RD_{MRDP1}, RD_{ne2} and RD_{MRDP2} are equivalent here and after. The minimum RD principle or extreme vector concept leads to a considerably simplified solution for RD_L. The RD of a curve can be solved provided two extreme vectors are found. Obviously, the extreme vectors are such two vectors between which there is the maximum included angle. Such vectors can be found by examining the normal vectors at each point on the curvilinear. For a given plane curve $y=f(x)$, its tangential equation at an arbitrary points p_i is

$$y\text{-}y_i\text{=}f'(x_i)(x-x_i) \tag{3-5}$$

and normal equation is

$$y\text{-}y_i\text{=}-\frac{1}{f'(x_i)}(x-x_i) \tag{3-6}$$

The normal vector at p_i can be written as

$$\mathrm{n}_i = (\text{-}f'(x_i),\ 1) \tag{3-7}$$

If $\boldsymbol{n}_i$ is represented in angle, then it can be written as

$$\boldsymbol{n}_i = (\alpha_i) \tag{3-8}$$

where α is the angle included between the normal and the positive direction of x-axis. Equation (3-7) and (3-8) are equivalent. Two extreme normal vectors can be defined as follows: The vector having maximum angle $\alpha_{\max}$ denotes $\boldsymbol{n}_{e1}$ and the vector having minimum included angle $\alpha_{\min}$ denotes $\boldsymbol{n}_{e2}$ respectively, as shown in Fig.3.4. The two extreme normal vectors present maximum included angle. They dictate the range of RD of the curve. The $\boldsymbol{n}_{e1}$ and $\boldsymbol{n}_{e2}$ on an arbitrary plane curve and corresponding RD map are shown in Fig. 3.4. RD_L is the intersection of RD_{MRDP1} (solid line) and RD_{MRDP2} (dashed line), representing the release direction of the curvilinear in x-y plane.

From Equation (3-1), the RD of MRDP_1 is given by

$$RD_{MRDP1} = \{\boldsymbol{d}\text{: } |\boldsymbol{d}, \wedge \boldsymbol{n}_{e1}| \le \pi / 2\} \tag{3-9}$$

and the RD of $MRDP_2$ is given by

$$RD_{MRDP2} = \{\boldsymbol{d}: |\boldsymbol{d},\wedge \boldsymbol{n}_{e2}| \le \pi/2\} \tag{3-10}$$

The initial angle of RD_{MRDP1} is equal to α_{e1}-π/2, i.e., α_{max}-π/2, the final angle of RD_{MRDP2} is α_{e2}+π/2, i.e., α_{min}+π/2. The RD of the curvilinear can be calculated by the formula

$$\mathrm{RD_L} = \{d_\alpha : (\alpha_{e1} - \pi/2) \le \alpha \le (\alpha_{e2} + \pi/2)\} \tag{3-11}$$

Obviously, the magnitude of α can be calculated by

$$\alpha = \pi - (\alpha_{e1} - \alpha_{e2}) \tag{3-12}$$

Let φ be the angle included between $\boldsymbol{n}_{e1}$ and $\boldsymbol{n}_{e2}$, then $\varphi = \alpha_{e1} - \alpha_{e2}$, the Equation (3-12) can be written as follows

$$\alpha = \pi - \varphi \tag{3-13}$$

Obviously,

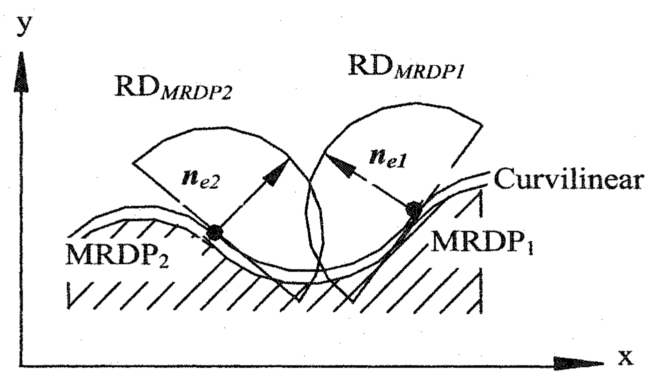
RD_L is a semi-circularity, i.e., $\alpha=\pi$, for $\varphi=0$;

RD_L is a single direction, i.e., $\alpha=0$, for $\varphi=\pi$;

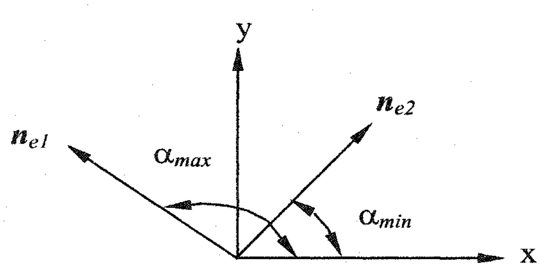
RD_L is between above two extremes, i.e., $0<\alpha<\pi$, for $0<\varphi<\pi$;

RD_L vanishes for $\varphi>\pi$.

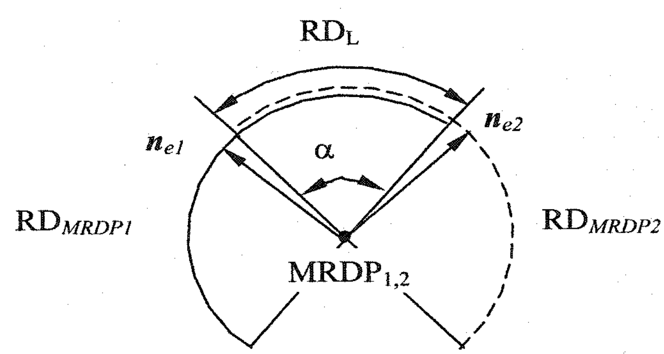
The first case shows the normal $\boldsymbol{n}$ of each point is identical with each other. The RD_L is equal to the RD of any individual point on the curve, which is a straight line case. The second case shows the angle included between the two normals equals π, such as hemi-circle, the intersection of RD*ps* is a single direction. The third case shows that the included angle between two normal vectors is between 0 and π, the RD_L is between the above extremes. The last case shows that when the included angle exceeds π, there is no RD intersection between the RD_{MRDP1} and RD_{MRDP2}. The curvilinear cannot be released in any direction in this case. Some particular examples are shown in Fig. 3.5, the RD map is illustrated in x-y plane.



(a) $MRDP_{1,2}$ on a plane curve

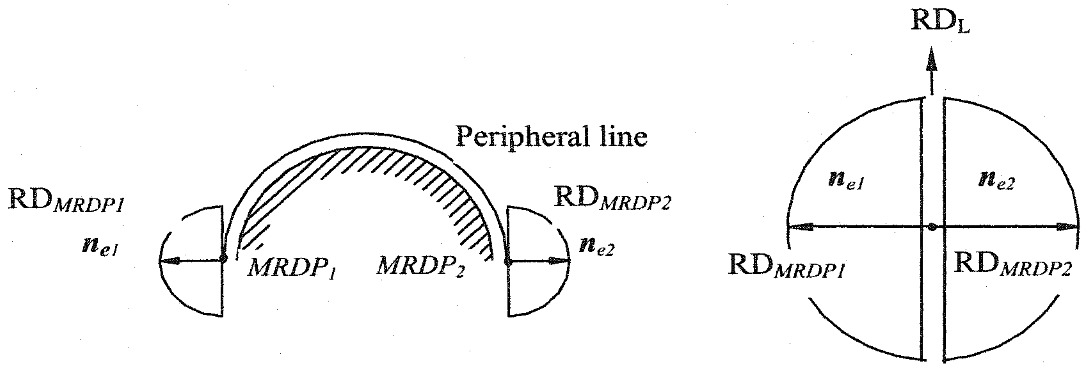


(b) Angle representation

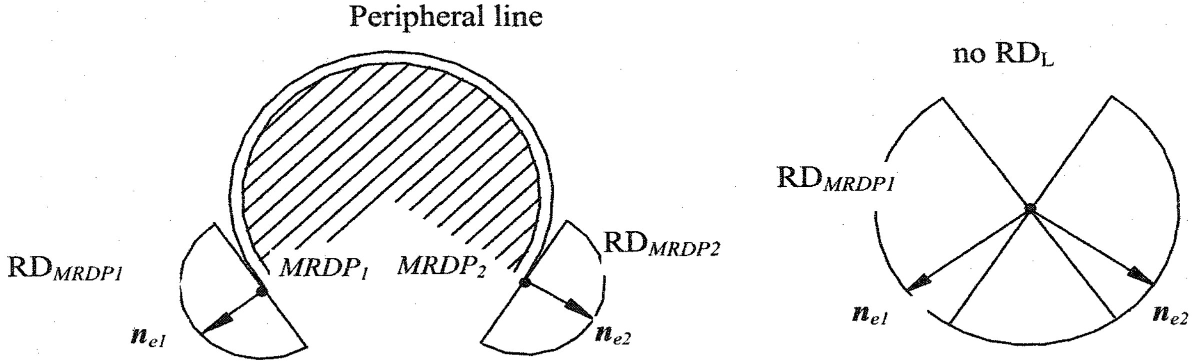


(c) RD map of $MRDP_{1,2}$

Fig. 3.4 RD of a plane curve



(b) Half circle case (φ=π)



(c) Greater half circle case (φ>π)

Fig. 3.5 Examples of RD for different curves

3.5 RD of a curved plane

A die casting is enveloped by rigid surfaces. The removal of a rigid surface is also the rectilinear motion of a rigid body. Similarly, a rigid surface can be removed from a die cavity in a direction or a set of directions, if and only if all the points on the surface can be removed in the direction or the set of directions. Let P be a curved plane on the casting surface. The RD of the plane, RD_P, is also determined by the intersection of RDs of all the points on the plane. A formula similar to RD_L can be given as follows:

$$RD_P = \{RD_{p1} \cap RD_{p2} \cap \ldots \cap RD_{pi} \ldots\} \tag{3-14}$$

It is still impossible to solve the intersection of RDps because of infinite points on a curved plane. Like the case of a curvilinear, if there are extreme vectors which are limited in number and can determine the RD_P, and the vectors can be found, the RD_P problem can be considerably simplified and can be solved through a small number of procedures. The following section will discuss the extreme vectors of a curved plane.

Suppose an arbitrary curved plane is defined by the function

$$F(x, y, z) = 0 \tag{3-15}$$

and suppose the function is continuous and monotropic in given domain. $p(x_p, y_p, z_p)$ is an arbitrary point on the plane. The tangential plane equation of the plane at point p is as follows

$$F_x(x_p, y_p, z_p)(x - x_p) + F_y(x_p, y_p, z_p)(y - y_p) + F_z(x_p, y_p, z_p)(z - z_p) = 0 \tag{3-16}$$

The normal vector of the tangential plane is

$$n_p = \{F_x(x_p, y_p, z_p), F_y(x_p, y_p, z_p), F_z(x_p, y_p, z_p)\} \tag{3-17}$$

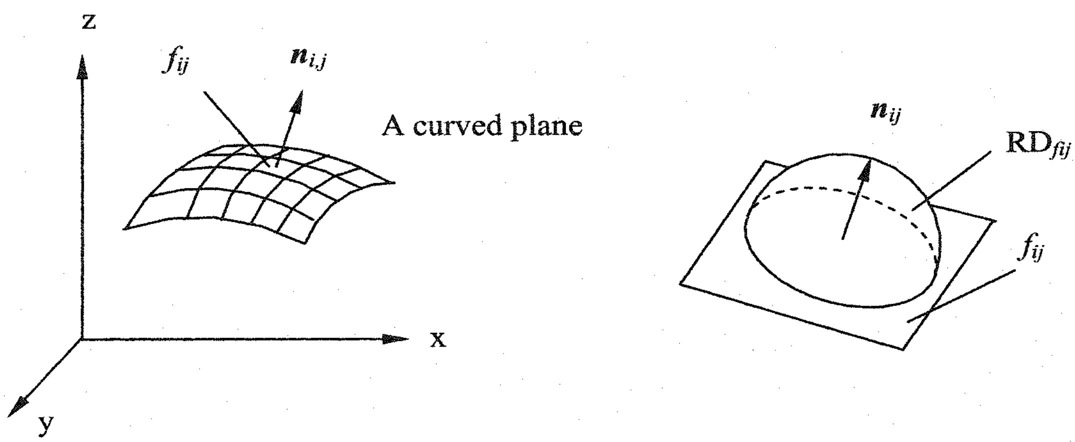
With the normal vector $\boldsymbol{n}$, the RD of each point on the plane can be uniquely determined by Equation (3-1). The RD of a curved plane is determined by intersection of the RD_ps defined by all normal vectors on the plane.

In the discussion of RD_L problem, a curved line is considered to be composed of a number of points. Extending the concept to a plane, a plane can be considered to be composed of a number of small face pieces. Let $P=\{f_i : f_i \in S\ (i=1, 2, 3, \ldots, m)\}$ represent a curved plane, f_i be the ith face piece on the plane. Thus, the equation for RD_P can be written as

$$RD_P = \{RD_{f1} \cap RD_{f2} \cap \ldots \cap RD_{fi} \cap \ldots \cap RD_{fm}\} \tag{3-18}$$

where RD_{fi} is RD of the ith face piece. Now a plane is divided into limited number of face pieces. The Equation (3-18) correlates RD_P with RD_f. The RD_f should be solved so as to solve for RD_P.

If the face piece is sufficiently small, it is appropriate to consider the piece to be a flat plane with normal $\boldsymbol{n}_i$ defined by Equation (3-17). The RD defined by the normal represents the RD of the entire piece. Thus a curved plane is discretized into a limited number of face pieces with respective normal vector and respective RD as shown in Fig. 3.6. The RD_P is determined by the intersection of a number of RD_f of small pieces. The minimum RD principle or extreme normal vectors are efficient ways for solving the RD of a curved line since the normal at any point on a plane curve is a plane vector. However, a curved plane is described by a function with multiple variables, the normal vector at any point on the curved plane is a space vector which cannot be defined by a plane angle. Suppose that extreme vectors exist in the case of curved plane. The question arises as to how these vectors are identified. In the following section, the concept of vector plane (VP) will be introduced for finding the extreme vectors.



(a) A discrete curved plane (b) RD of a face piece

Fig. 3.6 Discrete plane and RD of surface piece

If the normal vectors at all the pieces of a curved plane are translated into radius vectors, each vector of them would scatter from the coordinate origin along a certain direction in the form as shown in Fig. 3.6 (a). In fact, the intersection of RDs of all face pieces is only determined by the peripheral vectors since any pairs of vectors with the maximum included angle between them are bound to occur in peripheral vectors. The peripheral vectors are the extreme vectors required. Since the pair of extreme space vectors cannot be uniquely defined for space vectors, it is necessary to develop a general approach to determine all pairs of extreme vectors. A vector plane approach is proposed for this purpose. The peripheral vectors can be expressed in terms of vector plane. Let VPC denote all of the planes through the z axis, called plane cluster through the z axis, the equation of the plane cluster can be written as

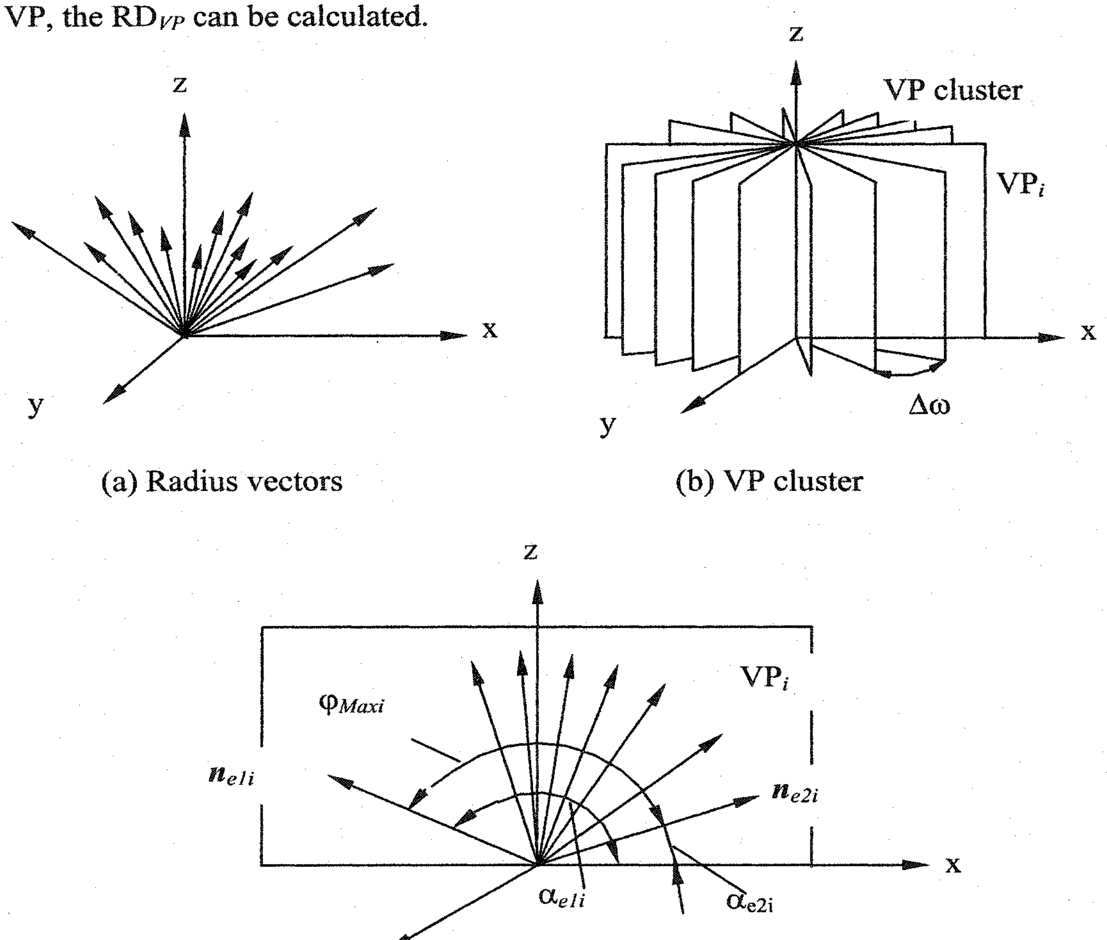
$$y = \tan(\omega)x \tag{3-19}$$

where ω is the angle included between a plane and the x axis. To every ω, there corresponds a plane through the z axis. If there are sufficient planes in the cluster, each vector shown in Fig. 3.7 (a) may lie in one of the planes. The cluster is called vector plane cluster (VPC) and each such plane in the VPC is called a vector plane (VP) as shown in Fig. 3.7(b).

Establishment of the VP makes it possible to find peripheral vectors and pairs of extreme vectors. When ω_0 is assigned to ω, the equation of VP corresponding to ω_0 is $y = \tan(\omega_0)x$. From Equation (3-17), the normal vector at any point on a curved plane can be defined. All the vectors with components in the x axis and the y axis satisfying the following relation

$$\frac{F_y(x,y,z)}{F_x(x,y,z)} = \tan(\omega_0) \tag{3-20}$$

will lie on the VP corresponding to ω_0. If an angle increment $\Delta\omega$ is specified, a number of VPs can be defined at angular intervals of $\Delta\omega$. Each vector of small faces on curved plane can be collected into a closest VP as shown in Fig. 3.7(c). Since the plane is discrete, vectors in a VP are limited in number. Besides, the space vectors are classified as plane vector-like, vectors in any VP can be operated in a similar way as in curvilinear. Like RD of a point, the RD_f is a hemisphere surface. For each VP, the α_{min} and α_{max} included between extreme vectors and x-y plane should be detected first. In general, there must exist at least a pair of extreme vectors between which the included angle φ is maximal in a vector plane unless the plane is a flat one. The two extreme vectors, referred to as $\boldsymbol{n}_{e1}$ and $\boldsymbol{n}_{e2}$, are the dominant vectors required. After the φ_{max} and two extreme vectors are found in a VP, the RD_{VP} can be calculated.



(a) Radius vectors (b) VP cluster

(c) A VP and vectors on the VP

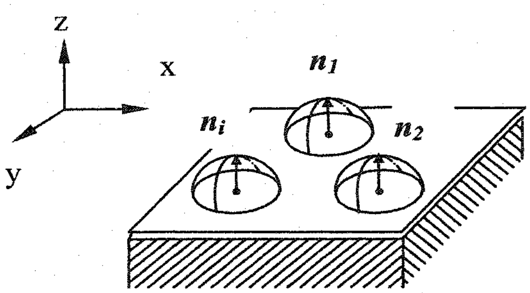
Fig. 3.7 VP and VP cluster

All the space vectors at each face piece on a curved plane are represented as vectors in a particular VP, RD_{VP} for each VP can be obtained. The RD of a plane, RD_P, can be solved by intersecting RD_{VP}s. Let VPC={ VP_i, $i=1,2,3...i,...n$} denote a cluster of VPs, RD_P of a given curved plane can be expressed as

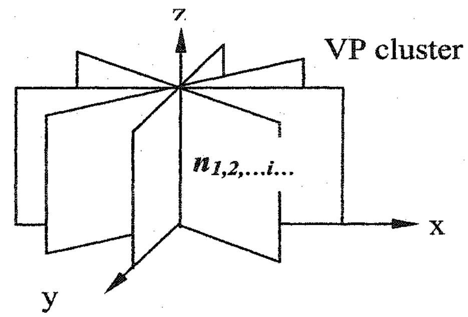
$$RD_P = \{RD_{VP1} \cap RD_{VP2} \cap \ldots \cap RD_{VPi} \cap \ldots \cap RD_{VPn}\} \qquad (3\text{-}21)$$

Each RD_{VP} can be obtained by intersecting RDs of two extreme vectors in the VP. Finally, the RD_P problem can be solved through limited number of procedures. It should be noted that the accuracy of the approach is related to the discrete size and the magnitude of $\Delta\omega$.

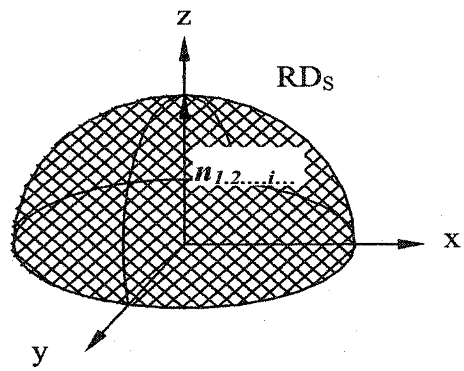
Some typical examples for RD_P are given in Fig. 3.8, Fig.3.9 and Fig. 3.10. Fig. 3.8 shows a flat plane case. Since the normal vector at each point on the plane is in the fixed upward direction, i.e., $\boldsymbol{n}_1= \boldsymbol{n}_2= \boldsymbol{n}_3, \ldots, \boldsymbol{n}_n$, all VPs share an identical vector. The RD_P in a flat plane case just equals the RD defined by a normal vector at any point p on the plane, that is, the RD_P for a flat plane is a hemi-sphere surface. Fig.3.9 shows a cylindrical sleeve case. Normal vectors on the cylindrical sleeve surface present a regular distribution perpendicular to the axis of the sleeve. There are two extreme vectors in a single VP. The RD_P is determined by a VP. Magnitude of RD_P is related to the size of segment of the sleeve. If the magnitude of the segment is represented by its circumferential angle φ which is also the maximum included angle φ_{max} between two vectors on the VP, then the RD_P of a cylindrical sleeve is dependent on the φ value. In the axial direction, the sleeve is freely released. The RD of a spherical surface depends on the magnitude of central angle of the surface. If the VPs are sufficient , the RD of a spherical surface is a spherical crown formed by a cone, as shown in Fig. 3.10.



(a) RD_p on a flat plane

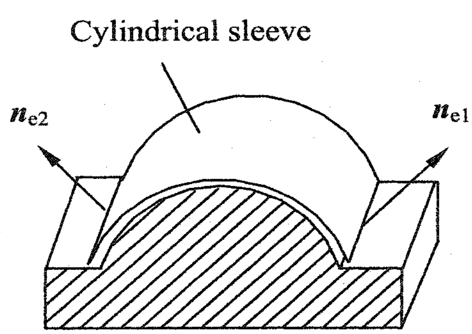


(b) VP cluster with identical vectors

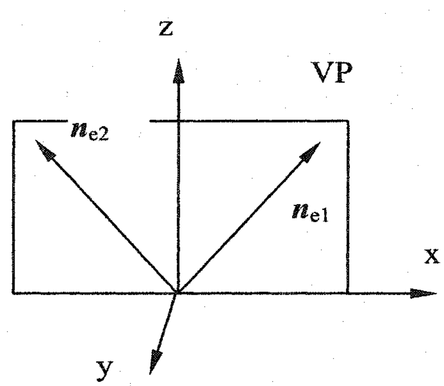


(c) RD map of a flat plane

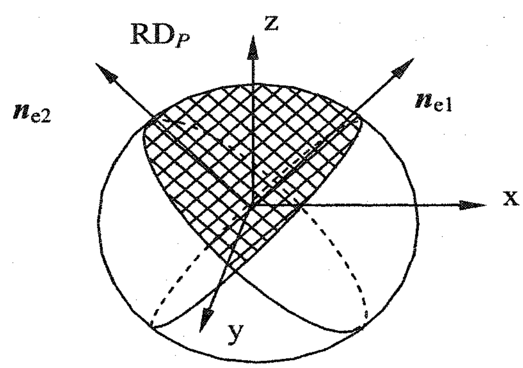
Fig. 3.8 VP and RD map of a flat plane



(a) A cylindrical sleeve

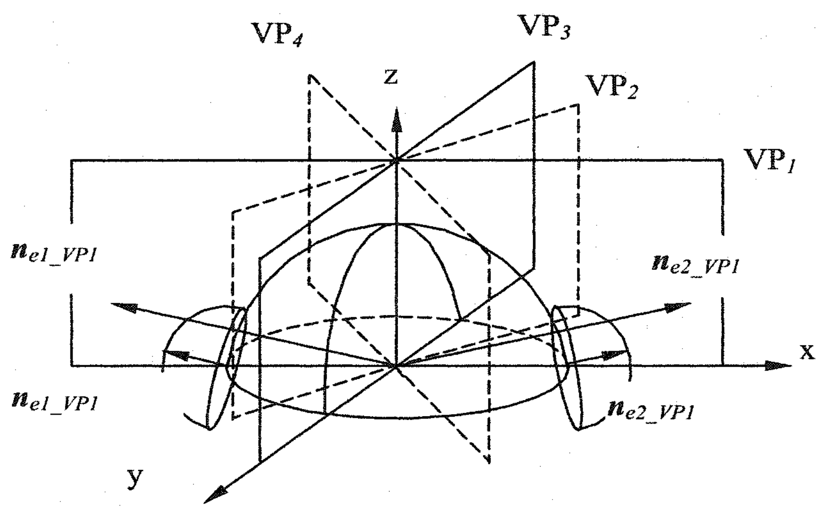


(b) VP and extreme vectors

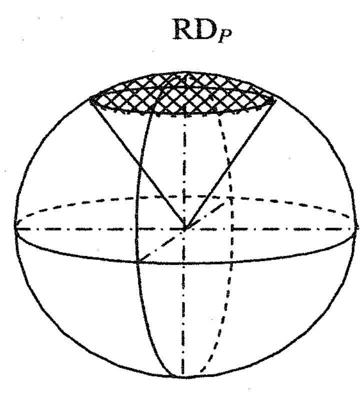


(c) RD map

Fig. 3.9 RD of a cylindrical sleeve



(a) VPs and extreme vectors



(b) RD map

Fig. 3.10 RD of a spherical surface

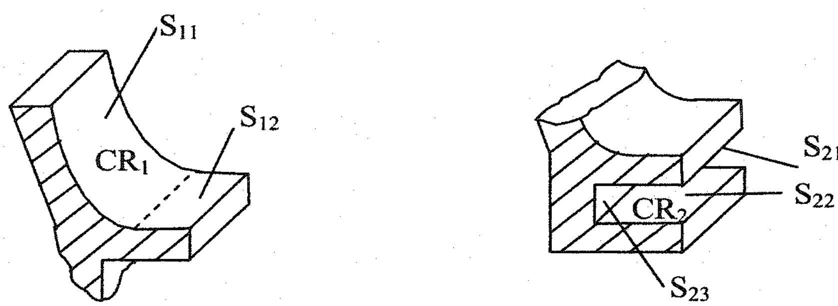
From the VP concept, the extreme vectors of general curved plane can be defined as the vectors at all points at the periphery, and the general curved plane refers to the spherical, ellipsoidal, parabolic surface, and cylindrical surface. The range of the RD of the various sphere is equal to the directions defined by the coning angle which is formed by tangents passing through each point at the periphery of the sphere to the central line of the sphere. The RD of cylindrical surface is dependent on the dihedral angle of two extreme tangential planes at the side edges of the cylindrical surface.

3.6 RD of a concave region

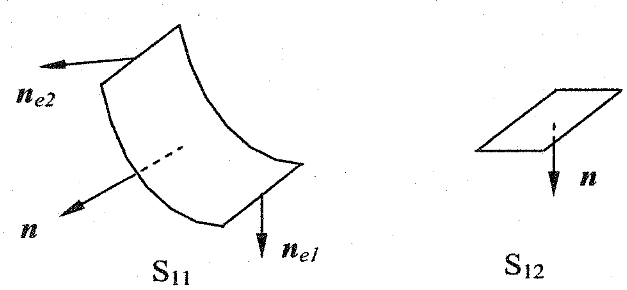
As discussed above, a CR is formed at least by a curved plane or two flat planes. In fact, the curved plane as a special CR case has been discussed in the last section. In this section, the general CR cases with multiple planes are discussed. The deduction for release of CR can be made as follows: A CR can be released from the die cavity in a direction or a set of directions, if and only if all surfaces in the CR can be released in the direction or the set of directions. Obviously, the RD of a CR equals the intersection of RD of all surfaces in the CR. Let CR={S_i, i= 1,2,…,i,…,n} denote a CR with n surfaces, and S_i is ith surface of the CR, then the RD of an CR, RD_{CR}, can be expressed as

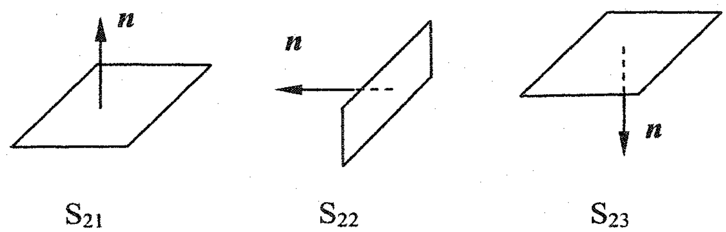
$$RD_{CR} = \{RD_{S1} \cap RD_{S2} \cap \ldots \cap RD_{Si} \cap \ldots \cap RD_{Sn}\} \tag{3-22}$$

The extreme vector approach described in the last section is efficient for determining the RD of each surface, such that the RD_{CR} can be solved. It should be noted that the RD magnitude is independent of the intersecting sequence in solving the RD of a CR. Each surface in the CR can be considered to be independent. Fig. 3.11 demonstrates the solving procedure of a RD_{CR}. Fig. 3.11 (a) shows two concave regions. The CR_1 is a two surfaces CR, that can be decomposed into two independent surfaces, S_{11} and S_{12}. S_{11} is a curved surface and S_{12} is a flat surface. The CR_2 is a three surfaces CR, it consists three flat surfaces, S_{21}, S_{22} and S_{23} with different orientations. Discomposed surfaces are shown in Fig. 3.11(b). The RD of each independent surface can be considered as an independent set of direction, therefore, it can be solved in the identical way as described in the previous sections. According to the Equation (3-22), the RD of a CR is determined by the intersection of RD of each surface in respective CR, as shown in Fig. 3.11(c). The contour of RD_{CR1} image is a part of spherical surface, and RD_{CR2} is a line in the RD map. It should be noted that the RD map is given in local pattern for the reason of simplification.

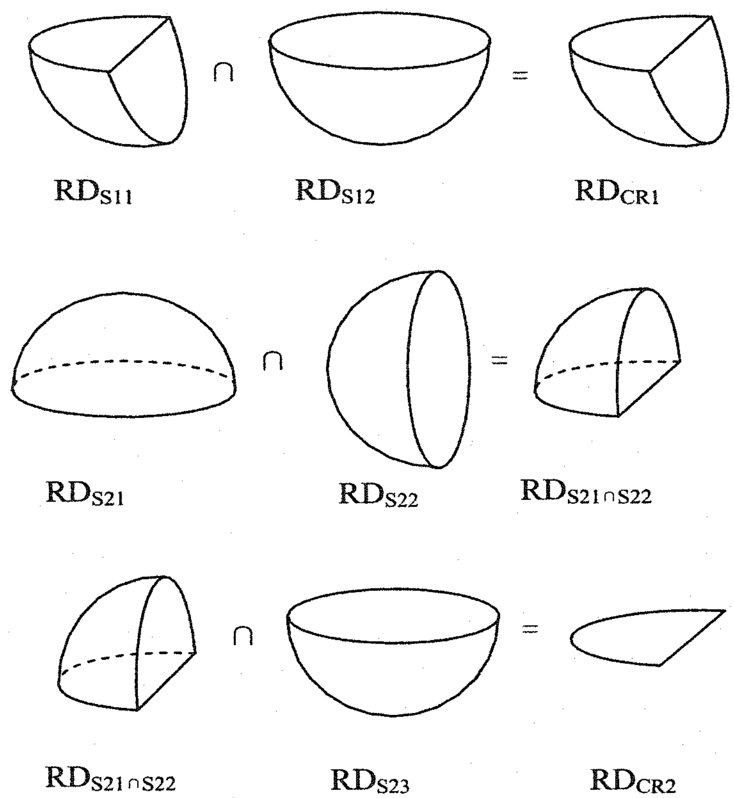


(a) Two CRs with different features





(b) Discomposed surfaces



(c) RD_S and RD_{CR}

Fig. 3.11 Demonstration of the solving procedure for RD_{CR}

The RD of a die casting is the intersection of RD of all CRs in the casting. Its solution will be further discussed in the next chapter.

3.7 Detection of a concave region

The purpose of feature analysis is to recognize concave regions and determine their RD in a die casting. The visibility approach developed by Chen et al [Chen 1993] is widely used to recognize pocket regions and extract convex hull [Weinstein and Manoochehri 1996]. The Boolean regularized difference operation between the moulded part and the core/cavity box is often used to extract the convex hull of concave feature with B-rep model [Fu et al 1999, Yin et al 2001]. These approaches are efficient but complex in application [Guleyupoglu et al 1994]. In this section, a three-dimensional ray detecting method is used for the recognition and extraction of concave region feature from a given casting model. In order to detect the concave region, a further definition for CR should be given with geometrical characteristics.

3.7.1 Definition of concave region

There may be a number of concave regions (CR) in a die casting. As discussed previously, the concave region can be a hole, slot, recess and undercut and so on. The concave region has the following geometrical characteristic:

The region is created by a surface or a number of adjacent surfaces, the dihedral angle of the adjacent surfaces, or the dihedral angle of the two tangential planes at two points at least on a surface, is less than 180 degree. Such regions are called concave regions.

According to the above definition, the formation of a CR must be one of the following three cases:

i) a curved surface at least;

ii) two adjacent flat surfaces at least;

iii) a combination of flat and curved surfaces.

The first case is indicated with a single surface S_{11} in Fig. 3.11 (b), the second case is CR_2, and the last is the case of CR_1 in Fig. 3.11 (a).

3.7.2 Detection of concave regions

A ray detection method is proposed to identify the CR. A die casting can be represented by finite difference model [Yuan 1994]. For a given casting object Q, define a cubic space domain Ω, and $Q \subset \Omega$, then the domain is divided into meshes. The die casting can be defined by a set of mesh points, $P=\{p : p \in Q \cap \Omega\}$. An attribute is assigned to each point in the set P so that the points can be recognized during detection of the CR. Let $p(i,j,k)$ be an arbitrary point outside the P and inside the domain Ω, semi-rays originated from p are emanated along six directions, i.e., x, -x, y, -y, z and –z, any ray may strikes against the cast body. Let CRI(p) be an array, called CR indicator, to record the total number of the rays which strike against the cast body. A point in a CR has at least two rays to hit on the casting body since a CR always lies at the inner corner of a bending area. Therefore, for an arbitrary point $p(i,j,k)$, if its CRI value is equal to or greater than 2, i.e., at least two rays emanated from the point strike against the cast body, such points are defined as CR points.

Surfaces including such points are defined as CR surfaces. It should be noted that there is a false count when a point just lies in the area over a declined plane as shown in Fig. 3.12. The value of CRI(p) from the same flat plane or same outer surface of a curved plane is only accumulated once. Since the rays emit in 3D directions, this approach can recognize any CR in a die casting.

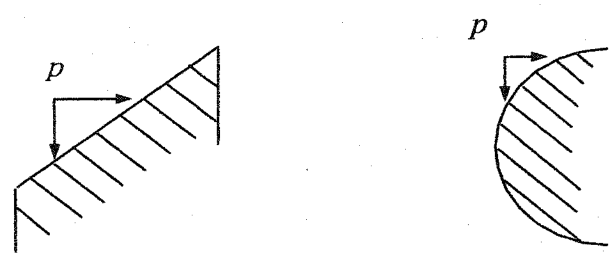


Fig. 3.12 False count in declined plane case and outer surface of curved plane

Fig. 3.13 shows typical points at different positions in the die casting shown in Fig. 3.2. For the point p_0, no ray emitted from it hits on the casting, CRI(p_0)=0. For the point p_5, only one of the six rays hits the casting, thus CRI(p_5)=1. From the definition of CR point, it is concluded that the points p_0 and p_5 are not CR points, that is, p_0 and p_5 are not inside any CR since the CRI value of points are less than 2. For the point $p_{2\text{-}4}$, there are at least two rays from each to hit on the casting. The CRI value of these points are equal to or greater than 2. The point $p_{2\text{-}4}$ are CR points, as they are inside a CR. The CRI value and the rays hitting on the casting from each point are listed in Table 3.1.

It should be noted that the CRI value may be different for different points in the same CR due to different positions where the point lie, and this does not affect the recognition of

CR. All the CR points can be recognized by examining the value of each point. In normal cases, different values of CRI reflect the different geometrical feature of CR. The CRI value of 2 or 3 corresponds to the CR of a through-slot case. The CRI value of 3 or 4 corresponds to the CR of a blind slot or a through hole. If the CRI value is equal to 5, the CR will be the case of a blind hole. A dead CR will occur if the CRI value equals 6. In this case, a split core may be required. The typical CRs with different CRI value are shown in Fig.3.14. The classification of CR will be discussed in the next chapter.

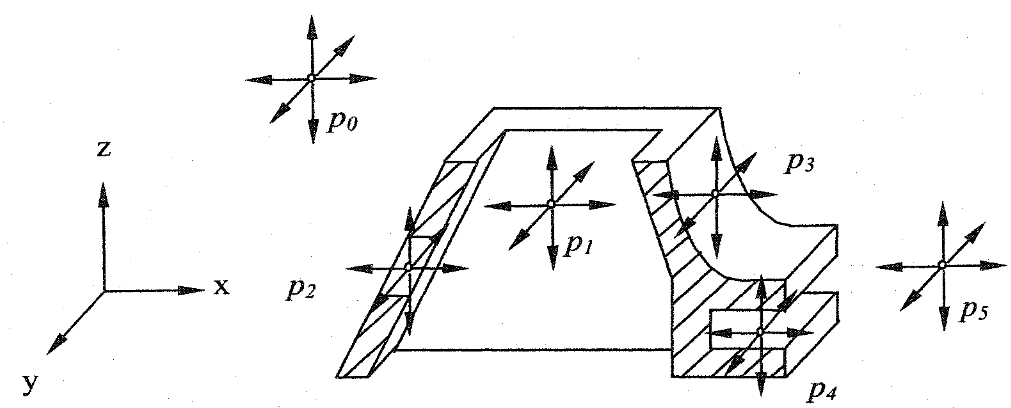


Fig. 3.13 Ray detection method and typical points

Table 3.1 The rays hitting on the casting and CRI value of each point

Point	Rays hitting on the casting	CRI value
p_0		0
p_1	x, -x, y, -y	4
p_2	x, y, -y, z, -z	5
p_3	-x, -z	2
p_4	-x, z, -z	3
p_5	y, -y, z, -z	1

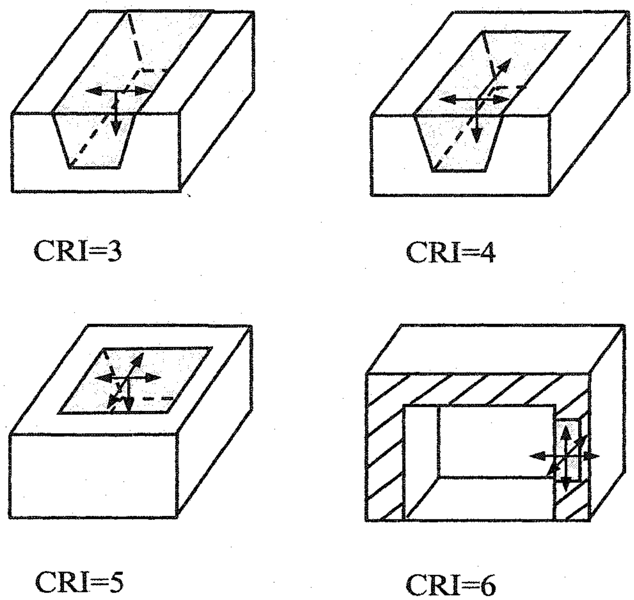


Fig. 3.14 The CR with different CRI value

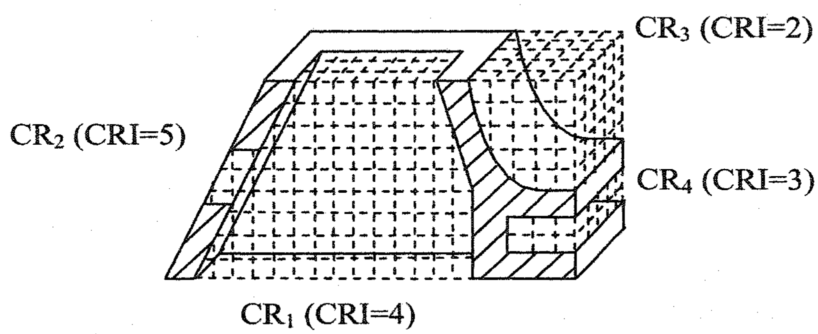
3.7.3 Extraction of concave regions

After a sweeping is made for the geometrical model of a die cast part, all the CR points can be recognized by examining the CRI value of each point in the domain. The CR can be thus extracted by storing and displaying all points with CRI value equal to or greater than 2. In fact, the interference between the part and die occurs on their surfaces during casting removal. The calculation of the RD only require the CR surfaces, it is sufficient to extract the CR surfaces to create CR hull. The set of points satisfying following two conditions are defined as CR surface:

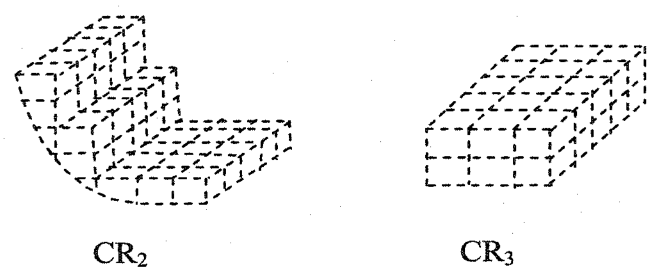
i). Points with CRI value equal to or greater than 2

ii). Points contacting with surface of the die casting

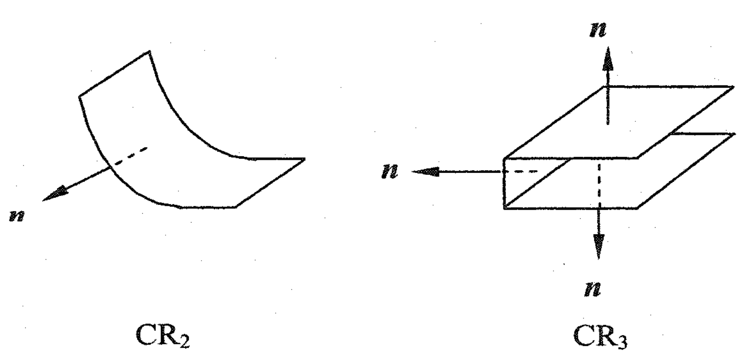
The points contacting with casting part can be extracted by the geometrical attribute of the points on a surface. In addition, the geometrical attribute of a point can also be detected since the data structure of die casting model carries the geometrical information. The die casting part shown in Fig. 3.13, for example, can be taken to illustrate the CR recognition and extraction. After recognition procedure, all points with CRI value equal to or greater



(a) CR points



(b) The points on CR



(c) CR shells

Fig. 3.15 CR recognition and extraction

than 2, i.e., CR points, are shown in Fig. 3.15(a). Then the CR points are examined and the points contacting with casting are extracted. Such points are shown in Fig. 3.15 (b). From these points, the CR shells can be determined as shown in Fig. 3.15(c). With CR shells extracted, the RD of each CR can be analyzed.

Based on the formation characteristics, the CR can be divided into two categories, i.e., simple CR (SCR) and complex CR (CCR). The SCR is formed by a single primitive, such as spherical, parabolic, cylindrical, cone surfaces and so on. The CCR is formed by intersection between primitives. In fact, the detection of the CR is mainly for CCR, the SCR can be presented during constructing the part model.

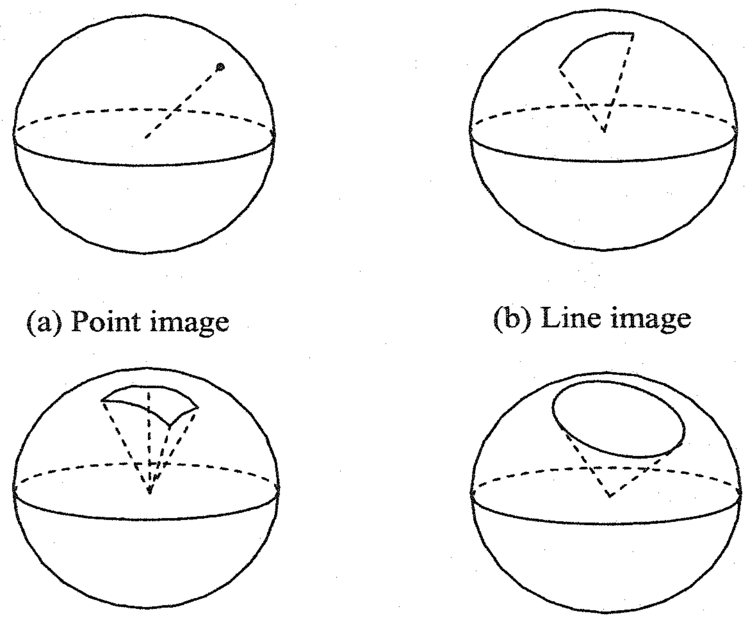
3.8 RD distribution and parting direction

3.8.1 RD distribution of a casting

In die design, the parting direction is defined to be a pair of opposite directions along which the die opens. The parting direction selection is important because it affects all the subsequent steps in die design, and involves many factors, such as number and volume of side cores, shape of parting plane, draw height, *etc*. Many researches select a direction as optimal parting direction in which the number of side cores is minimal [Lu and Lee 1998, Fu et al 1999, Yin et al 2001]. Therefore, the distribution of RD is necessary for the determination of the use of side cores.

Given a die casting, there may exist a number of CRs in it, each of them corresponds to a specific RD_{CR}, a direction or a set of directions to release the CR. Understanding of RD_{CR} distribution is important for parting scheme design. The RD distribution refers to the frequency of RD_{CR} occurring in each direction. Suppose that a die casting with n CRs is given, the direction or the set of directions in which the CR_i is released is denoted as RD_{CRi}, which can be computed by using the algorithm discussed in the previous section. The distribution of RD can be observed by the RD image on a RD map. The RD_{CR} mapping onto an unit spherical surface is called the RD image. Since each RD_{CR} corresponds to a particular image, a set of RD images is obtained depending the type and number of CR. In general, there are four types of RD images, that is, point, line or bar, polygon and round image. A point image represents an unique direction, a line segment represents a set of directions bounded in a sector plane. A polygon or a round image represents a set of directions bounded in a pyramid or cone as shown in Fig. 3. 16.

Since the RD_{CR} and its image are equivalent, the set of RD images is similarly denoted as RDS_{CR}={RDS_{CR1}, RDS_{CR2}, ···, RDS_{CRn} }. There are four cases for any two images, i.e., congruence, inclusion, intersection and separation, as shown in Fig.3.17. For case (a), the two images completely overlay, this indicates the two CRs forming the two RD images have identical range of release direction. If any direction in the image is selected as parting direction, the two CRs can be released. Case (b) indicates that any direction in the shadow area can release the two CRs forming the two images, but the

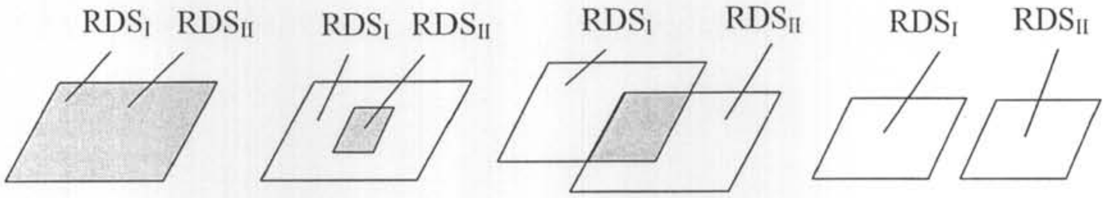


(a) Point image (b) Line image

(a) Polygonal image (a) Circle image

Fig. 3.16 Four typical RD images

direction out of the shadow area can release only CR_I, and CR_{II} cannot be released. In the third case, there is an intersection between image I and image II. For the directions in the intersection, two CRs can be released. Out of the intersection, only one CR, CR_I or CR_{II}, can be released depending on area of the selected direction. In the last case, any direction in two separate images is selected as parting direction, only one CR can be released depending on the image of selected direction. There is no RD shared by two CR_S. In this case, at least one side core should be required. In fact, the purpose of resolving RD distribution is to understand the RD image position and relationships between RD images. Solution of the RD distribution can be formulated as a covering problem of RD images given above. There is, however, no existing algorithm available for solving such problems. A function algorithm was introduced by Sun [Sun and Yang 1995] to examine the planar covering problem. The algorithm requires the describing function of edges and vertexes of planar polygons. Such an algorithm is tedious and not always efficient for the present problem since the describing function of RD images cannot be always obtained. A greedy heuristic approach was suggested by Chvatal [Chvatal 1979] and was employed by Chen [Chen 1992]. Based on the works of Chen, an algorithm was developed for planar surface by Weinstein [Weinstein and Manoochehri 1996]. In this study, an accumulation function is introduced to determine the RD distribution.



(a) Congruence (b) Inclusion (c) Intersection (d) Separation

Fig. 3.17 Four relationships for any two images

Let CR={CR$_i$, i=1, …, n} denote the set of CRs in a given die casting, then RDS$_{CR}$={ RDS$_{CRi}$, i=1, …, n} denotes the set of RD image in the RD map. Based on the minimal use of side cores, the parting direction should be a direction in the direction domain Ω, Ω={RDS$_{CR1} \cup$ RDS$_{CR2} \cup \cdots \cup$ RDS$_{CRn}$}. For each direction or point in the direction domain Ω, a test is made to determine the frequency the direction is covered by RDS$_{CR}$. The frequency will be recorded by an accumulation function, called the RD distribution factor (RDDF). This procedure is repeated until all directions or points in Ω are swept. Given a direction Ω_j, $\Omega_j \in \Omega$, then the RDDF in the direction Ω_j can be written as

$$\mathrm{RDDF}(\Omega_j) = \sum_{i=1}^{n} \mathrm{RDDF}_{\mathrm{CR}\,i} \tag{3-23}$$

where

$$\mathrm{RDDF}_{CRi} = \begin{cases} 1 & \text{if } \Omega_j \in \mathrm{RD}\,S_{CRi} \\ 0 & \text{otherwise} \end{cases}$$

In fact, the accumulation function records the number of CRs which can be released in each direction. Taking the casting shown in the Fig. 3.15 for an example, its RD map is illustrated in Fig. 3.18. There are four RD images, RDS_{CR1}, RDS_{CR2}, RDS_{CR3} and RDS_{CR4} in the RD map. The four images, corresponding to CR_1, CR_2, CR_3 and CR_4 respectively, create four RDDF areas, denoted as area *I*, area *II*, area *III* and area *IV*, which present different RDDF values. The image of RD_{CR1} is a spherical polygon. Intersection of RDS_{CR1} and RDS_{CR3} creates area *I*, that is, the direction points in the area *I* are covered in twice by RDS_{CR1} and RDS_{CR3}. It should be noted that the RDDF calculation includes the antipodal image (area of dot-dashed line). The RDDF value in this area is equal to 2. The RDS_{CR2} image is a point, a single direction formed by CR_2. The point is covered by RDS_{CR2}, RDS_{CR3} and RDS_{CR4}. Area *II* is a point, the RDDF value in Area *II* is 3. Intersection between RDS_{CR3} and RDS_{CR4} is a spherical line, i.e., area IV, the RDDF value in this area is 2. Area *III* (deduct the antipodal image of RD_{CR1}) is single direction area, the RDDF value is equal to 1 in this area. The RDDF in a direction is equal to the number of times that the direction point is covered by the RDS images. Thus the RDDF for each direction can be obtained by sweeping all defined direction points and by counting the number of times covered by all the RDS images.

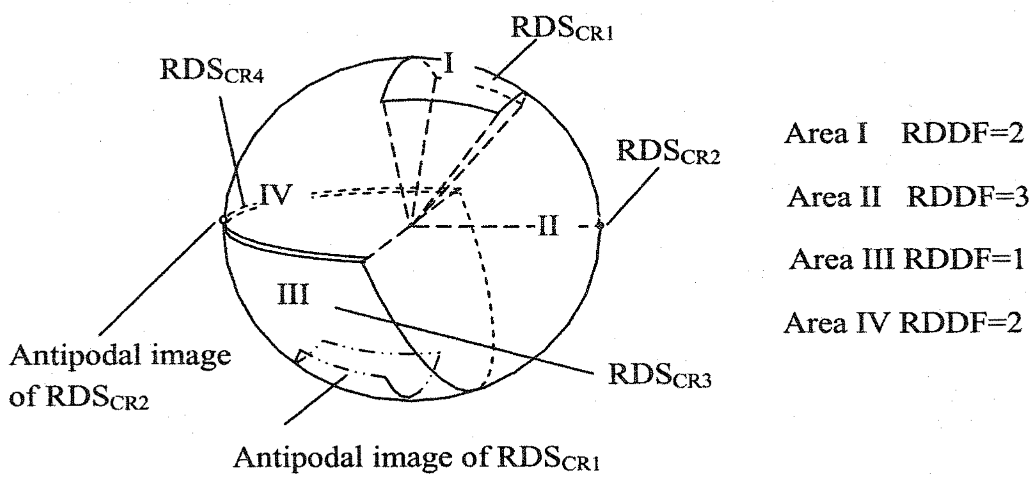


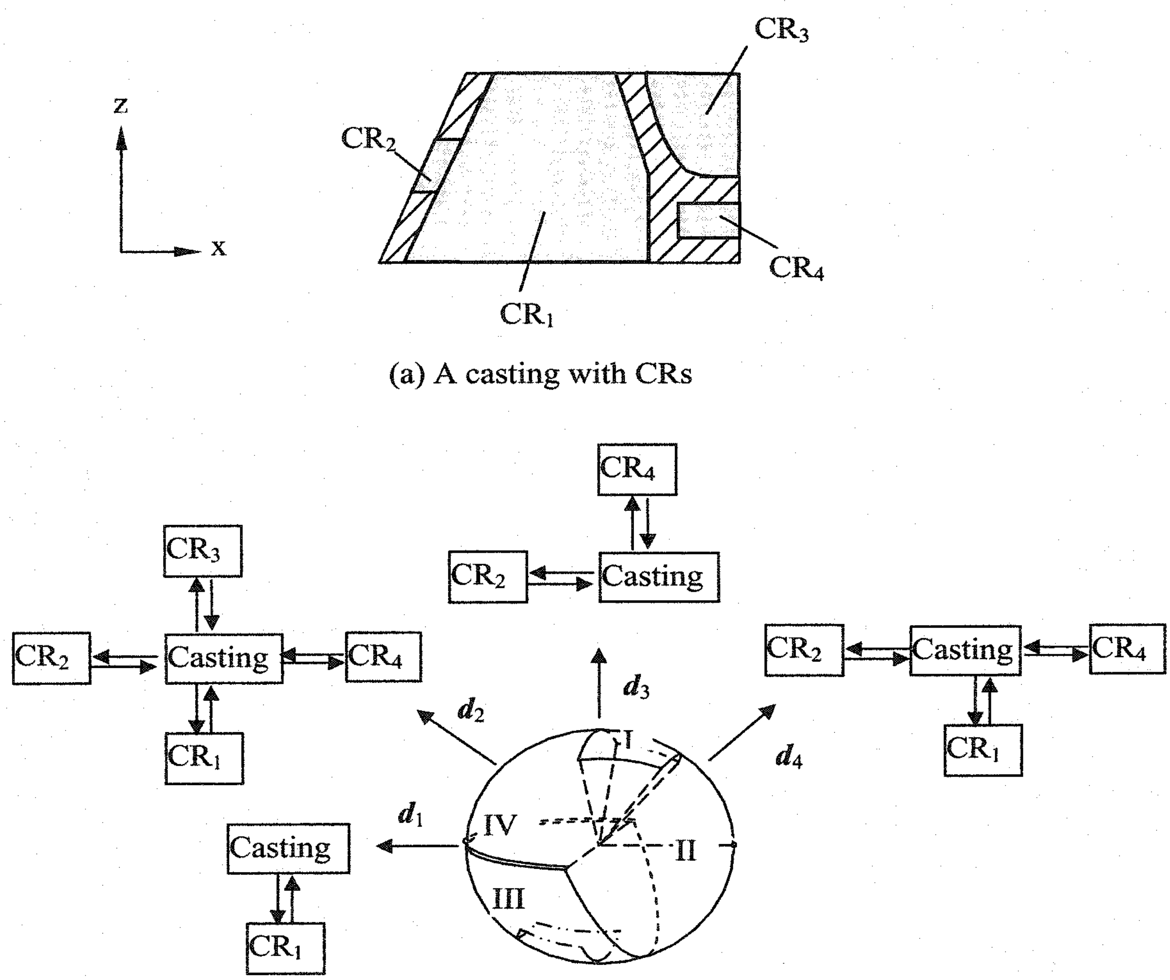
Fig. 3.18 RD_{CR} distribution and RDDF value in different areas

3.8.2 Influence of RD distribution on parting direction

In principle, any direction can be used as the parting direction of a die casting. However, due to requirements of a simplified die and favorable filling condition, particular specifications, there is only a limited number of directions suitable for serving as parting direction. An important task in die design is to determine the direction or the set of directions. As discussed earlier, the use of side cores increases the manufacturing cost, complicates the casting operation and slows down the casting process. From this consideration, the parting direction with minimum use of side cores would be preferred. In general, the number of cores is equal to that of CR. This can be recognized by the CR detection procedure. For CRs which cannot be released in a given parting direction, the side cores are required. The number of side cores induced for each possible parting direction should be understood for a parting scheme design. The RDDF presents such information. The distribution of RD influences the selection of the parting direction, and the influence can be observed by a directional interference graph. Consider a die casting part with all CRs, as shown in Fig. 3.19 (a), there are four CRs in the casting, CR_1, CR_2, CR_3 and CR_4. The RD distribution and directional interference graph is shown in Fig. 3.19 (b). In the directional interference graph, the arrow lines represent the interference between the casting and the CR in a given direction. After the casting is given, the RD distribution is created. The distribution of the RD influences the use of side cores and the selection of the parting direction. In the given four directions, there are different interference cases. The CR_1 prevents the casting from removing in the direction d_1. CR_2 and CR_4 prevent the casting from removing in the direction d_3, CR_1, CR_2 and CR_4 prevent the casting from removing in the direction d_4. In direction d_2, all four CRs prevent the removal of the casting, and four side cores would be required if the direction is selected as parting direction. In general, there exist a number of image areas with different RDDF in RD map, representing the RD distribution of CR. The RD distribution provides the relationship between the parting direction and the use of side cores. The problem of determining preferred parting direction is equivalent to that of finding an image area Ω_m with the maximal RDDF.

Consider a direction $\boldsymbol{d}$ ($\boldsymbol{d}^+$ and $\boldsymbol{d}^-$), $\boldsymbol{d} \in \bigcap_{i=n} RD_{CRi}$, since the direction $\boldsymbol{d}$ is shared by n RD_{CR}, i.e., $\boldsymbol{d}$ is in the intersection of n RDS_{CR}, then n corresponding CRs can be removed from a die cavity if the $\boldsymbol{d}$ is used as parting direction. It implies that the cores forming the n

CR can be designed to fixed one and incorporated into cover die or ejector die. Such CRs do not cause any interference for removal of a die casting from the die. Any directions out of the intersection should cause interference for the removal. These are interference elements (IE), and require use of side cores to release them. Particularly, a casting can be removed from a die cavity without use of any side core if a direction, which is included in the intersection of all RDS, is used as parting direction.



(a) A casting with CRs

(b) RD distribution and directional interference graph

Fig. 3.19 Directional interference graph

A different number of side cores can be used depending on the selection of parting direction. As shown in Fig. 3.19, two CRs, CR_1 and CR_3, can be released by fixed cores if the parting direction is chosen in Area I. Three CRs, CR_1,CR_2 and CR_3, can be released if the parting direction is chosen in Area II. CR_3 and CR_4 can be released along directions in area IV. If a direction in area III is selected, only CR_3 can be released. If the parting direction is chosen out of the four areas, no CR can be released by fixed core. The RDDF value indicates the number of CRs that can be released by fixed core. In general, the direction with high RDDF value has the priority in the selection of parting direction.

In order to obtain the information on which CRs can be released in a selected direction, an ownership vector can be used [Chen 93]. Each point Ω_j on the Ω can be assigned an ownership vector u(Ω_j), where

$$u(\Omega_j)=(u_1(\Omega_j), u_2(\Omega_j), \ldots, u_m(\Omega_j)) \qquad (3\text{-}24)$$

and

$$u_i(\Omega_j) = \begin{cases} 1 & \Omega_j \in RDS_i \\ 0 & \Omega_j \notin RDS_i \end{cases}$$

The vector $u(\Omega_j)$ keeps track of the RDS covering p. Two opposite point Ω_p and Ω_q are equivalent if $u(\Omega_p)=u(\Omega_q)$. Since the RDSs covering a point are recorded by its ownership vector, CRs released can be found by traversing the ownership vector for each covered point.

The analysis of the geometrical feature and release direction provides the foundation for parting scheme design. It can give all the possible parting directions and the number of side cores required in each possible parting direction. However, as the parting scheme design involves many factors, such as the size of the cores, the shape of parting plane, the draw height and so on. It is not sufficient for parting scheme design to be only based on the feature analysis. An optimal design of the parting scheme requires the combination of feature analysis and design rules. In the next chapter, a feature-based parting scheme design will be described.

3.9 Summary

In this chapter, the following is discussed:

i) Based on the mechanics principle, the basic characteristic of removal of a die casting is discussed. Terms used in feature analysis are defined and the methodology of feature analysis for die casting is described.

ii) Based on the translation characteristic of a rigid object in mechanics, the release direction of die cast objects is deduced. The release direction of various objects is analyzed. The extreme vector and vector plane approach are proposed for the determination of the release direction of arbitrary concave regions.

iii) A ray detection method is employed for recognizing and extracting concave features from a die casting. With this ray detection method, each concave features is assigned a block scalar. Any concave region and its hull can be built by examining the block scalars.

The points in concave hull carries geometrical information required for the determination of the normal vector and the release direction.

iv) An accumulation function is proposed to record the release distribution. The relationship between the release direction distribution and the parting direction is discussed. The foundation for parting scheme design is presented

CHAPTER 4

GENERATION OF PARTING SCHEMES

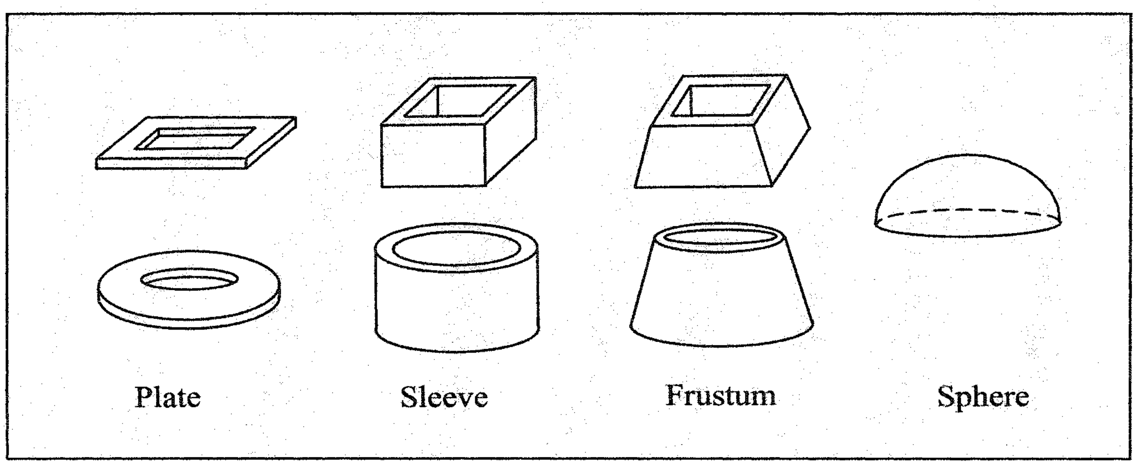
4.1 Basic description

As previously discussed, parting scheme design includes the determination of parting direction and the location of parting plane. Parting direction refers to the direction along which the die opens, and parting plane is the interface between two die halves. Parting scheme design is the first procedure in die design. Parting scheme dominates the die structure and all the subsequent steps. Parting scheme design requires a thorough analysis on geometrical features of the cast part and comprehensive consideration of design rules. From this viewpoint, parting scheme design is geometrical feature and design rule dependent. Therefore, the feature and rule-based approach are used for parting scheme design. The design process is divided into three steps: scheme generation, scheme evaluation, and scheme optimization. In the scheme generation stage, the concave region (CR) and release direction (RD) are used as basic features for parting direction selection, and the planes with maximal projected area viewed from each specific parting direction are used as parting plane. The scheme evaluation is performed according to design rules which are formulated as computable models. Based on the results of evaluation stage, an optimal scheme can be determined. In this chapter, the feature-based parting scheme generation is discussed.

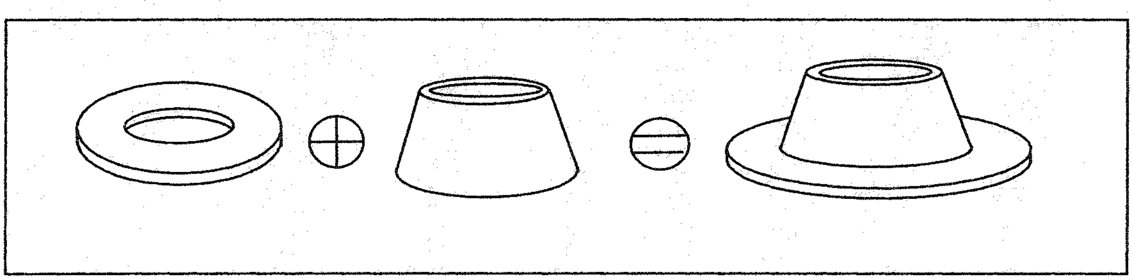
4.2 Modelling of the die casting part

In order to analyze the geometrical feature, a solid casting is required. The casting is modelled by the approach of constructive solid geometry. A casting part can be decomposed into a number of simple primitives. In other words, a casting can be composed of a number of basic geometrical features or primitives. If sufficient primitives are given, an arbitrary casting part can be constructed.

In this study, the four categories of basic primitives are defined, these are plate, sleeve, sphere, pyramid and frustum, as shown in Fig 4.1 (a). By prescribing the topological relationship between the primitives, a solid casting can be defined as shown in Fig 4.1 (b).



(a) Four categories of basic primitives



(b) A casting composed of two primitives

Fig. 4.1 Primitives and casting

"+" denotes the topological relation between the primitives and "=" denotes the formation of a casting.

4.3 Fundamental features

The conception of feature varies in different studies. In geometrical modelling, a feature is a geometrical primitive such as a cylinder, a cone, or a cuboid etc [Graham 1986]. In die casting design, the features are defined as a particular set of geometrical entities which impart the product function or the manufacturing process. A feature is any aspect of the part to which the designer must pay extra attention when designing castings, such as ribs, fillets, drafts etc [Liou 1990]. In parting scheme design, the feature is considered as all geometrical characteristics which influence the selection of a parting scheme. The parting scheme design includes the determination of the parting direction and the parting position. Therefore, there are two types of features concerned in parting scheme design, namely, the feature effecting the parting direction and the feature effecting parting position. As previously discussed, CR and RD influences the determination of parting direction, all the CRs and RD in a die casting are considered as the feature influencing parting direction. The recognition and extraction of the CR, and calculation of the RD have been discussed in the last chapter. Since die castings vary in geometry, the type and number of the CR in a die casting also varies. If the CR features are classified based on the CR characteristic, the identification of CR and RD could be simplified. In the following section, the classification of the CR is discussed.

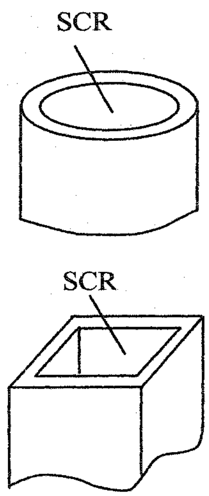
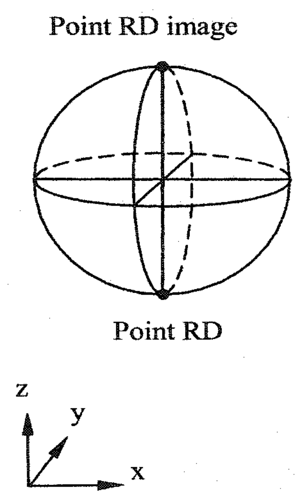
According to the formation of the CR, as discussed in the last chapter, there are two basic classifications of CR features, i.e., simple CR (SCR) and complex CR (CCR). The SCR is formed by a single primitive. The CCR is formed by intersection of at least two primitives. The classification of CR is needed for both identifying the CR and for calculating the RD of a die casting. When the predefined CR feature is recognized, the extreme vectors attached to each face of CR feature can be rapidly obtained.

4.3.1 Simple CR (SCR)

Based on the primitives defined in Fig. 4.1, the primitives forming the SCR include rectangular sleeve, cylindrical sleeve, pyramid frustum, cone frustum and sphere crown. As discussed earlier, there are four RD images for SCR, i.e., point, polygon, circular and line or bar image. As shown in the Table 4.1, the point image is formed by sleeve primitive. The feature of this type of SCR is a hole with a fixed cross section size. The geometry of the cross-section may be circular, rectangular and so on. The RD corresponding to this feature is always only a pair of opposite directions along the axis of the feature. Its image is a point on the RD map.

The polygon image is formed by a pyramid frustum. The feature of this SCR is a rectangular hole with variable cross-section. A through or blind hole comes with monotonic variable cross-section from the top. The magnitude of the RD, α and β, is dependent on the dihedral angle between two opposite surfaces. The circular image is created by a cone frustum and sphere, and the magnitude of the RD depends on the coning angle for cone frustum, or minimal coning angle formed on a sphere. The cone on a sphere is formed by tangents passing through each point at the periphery of the sphere to the central line of the sphere. The line or bar image is formed by a curved plane equal to or less than a half cylinder. Its RD is dependent on the magnitude of the central angle of the cylinder surface. The RD and characteristic corresponding to each SCR are given in Table 4.1.

Table 4.1 The RD corresponding to each SCR

CR cases	Characteristic	Release directions
 SCR forming point RD image	$\boldsymbol{n}_{CR(fi)} \cdot n_{Axis} = 0$ $\boldsymbol{n}_{CR(fi)}$ is the normal vector of each face in SCR, and n_{Axis} is the vector in axial direction.	

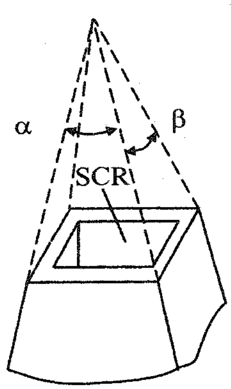
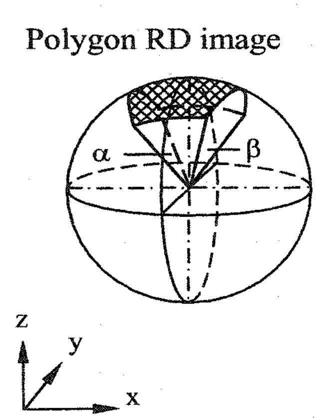
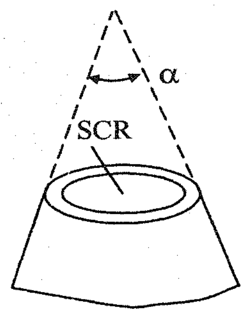
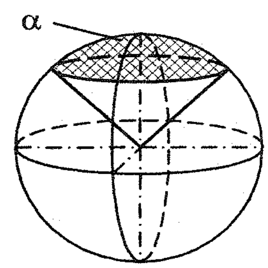
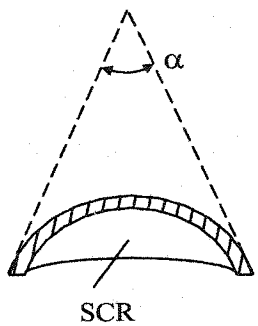
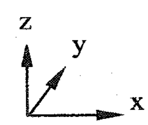
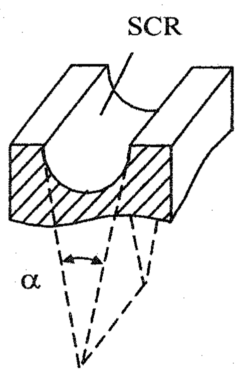
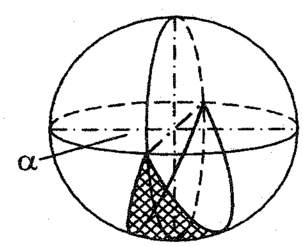
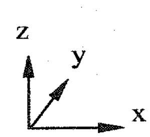
CR cases	Characteristic	Release directions
 SCR forming polygon RD image	$\boldsymbol{n}_{CR(fi)} \cdot n_{Axis} \neq 0$ α and β are the dihedral angles between opposite faces	

Table 4.1 The RD corresponding to each SCR (continued)

CR cases	Characteristic	Release directions
	α is the coning angle.	Circular RD image 
	α is the minimal coning angle formed on sphere.	
SCR forming circular RD image		

CR cases	Characteristic	Release directions
	α is the dihedral angle between two extreme tangential planes on cylindrical surface	 Bar RD image 
SCR forming bar RD image		

4.3.2 Complex CR (CCR)

The CCR refers to the CR formed by intersection of two or more primitives. Based on the number of faces required for defining RD of the CCR, the CCR can be divided into two-face CCR, three-face CCR and so on. As shown in Table 4.2, the two-face CCR refers to the RD of the CCR that can be determined by two faces in the CR. The magnitude of the RD, α, is determined by two extreme vectors at the two faces or by the dihedral angle of two faces. The corresponding RD image is a range less than half sphere. The three-face CCR refers to the RD of which can be determined by three faces in the CR. The magnitude of the RD, α and β, is dependent on three vectors at the three faces or on two dihedral angles among the three surfaces connecting one another. The image on the RD map is a triangular spherical region. A ring CCR is formed by one or two plate primitives and a sleeve primitive or a frustum primitive. It cannot be released if the ring angle is larger than 180° for two plate primitives and a frustum primitive. In this case, a split slide block is required or it is split by two die halves. The RD and characteristic corresponding to each CCR are given in Table 4.2.

Table 4.2 The RD corresponding to each CCR

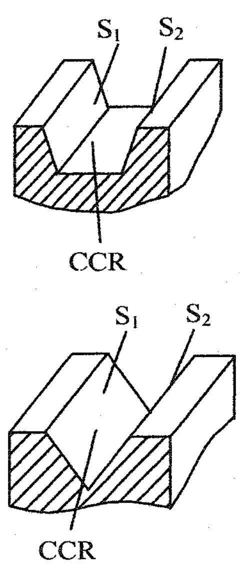
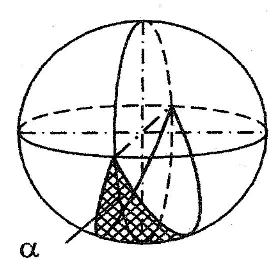
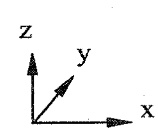
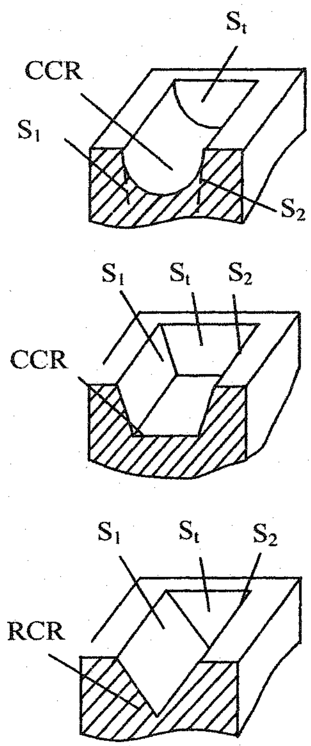
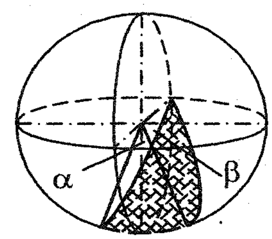
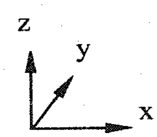
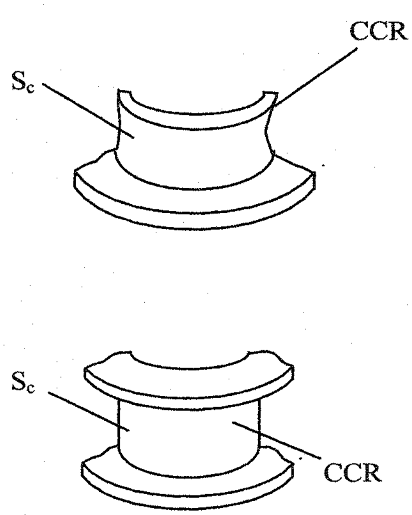
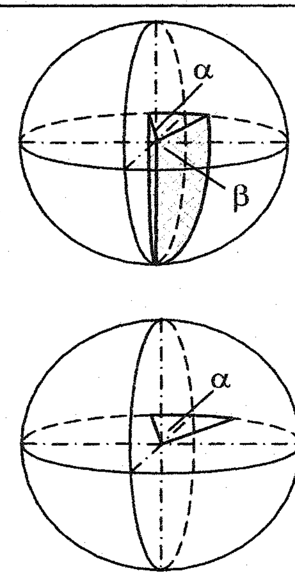
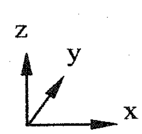
CR cases	Characteristic	Release directions
 Two face CCR	α is the dihedral angle of S_1 and S_2.	 Two face RD image 

Table 4.2 The RD corresponding to each CCR (continued)

CR cases	Characteristic	Release directions
 Three face CCR	α is the dihedral angle of S_l and S_2 and β is the ascent angle of S_t.	 Three-face RD image 

CR cases	Characteristic	Release directions
 Ring CCR	α is the central angle and β is the ascent angle of Sc. α is the central angle of Sc.	  Ring RD image

4.3.3 Parting plane feature

The parting plane is another feature for parting scheme design. Generally, a parting plane is always located at the plane with maximal projected area along the parting direction. Parting position is dependent on the parting direction and the primitives included in the parting plane. In general, the parting position coincides with the outermost points of a casting entity when viewed along the parting direction [Nainy et al 1997]. A die casting comprises one or several primitives, each entity presents a or a set of specific parting positions depending on the parting direction. The edges forming the projected boundary along the parting direction are considered to be the position of the parting plane. The feature for parting position can be determined for typical primitives as shown in Table 4.3.

The parting position is related to selected parting direction. For a plate primitive, the parting plane is located in position I if the normal of the plate is not perpendicular to the parting direction, i.e., $\boldsymbol{n}_{\text{plate}} \bullet \boldsymbol{d} \neq 0$. Otherwise, $\boldsymbol{n}_{\text{plate}} \bullet \boldsymbol{d} = 0$, the parting plane is in the position II. For sleeve primitives, there are also two ways to place a parting plane. When the axis direction of the sleeve coincides with parting direction, $\boldsymbol{n}_{\text{axis}} \bullet \boldsymbol{d} = 1$, the parting plane can be placed anywhere along the axis of the sleeve within the bounds of the upper and lower loop faces. When the axis direction of the sleeve does not coincide with parting direction, $\boldsymbol{n}_{\text{axis}} \bullet \boldsymbol{d} \neq 1$, the parting plane can be placed at the plane passing through an axis which has a maximum area of cross section. In this case, interference will occur during removal of the sleeve primitive, and a side core would be required. In fact, the $\boldsymbol{n}_{\text{axis}}$ is the RD of the sleeve, which is a single direction, or a pair of opposite directions. For frustum and sphere primitives, the position of the parting plane is related with the coning angle α. As discussed previously, for the case of a sphere, the cone is formed by tangents passing through each point at the periphery of the sphere to the central line of the sphere. The RD of frustum and sphere primitives is the directions defined by the coning angle. If the parting direction is included in the RD of each given primitive, $\boldsymbol{d} \in \boldsymbol{d}_\alpha$, the parting plane is located at the bottom face with maximum projection area. In this case, the fixed core can be used to release the primitives. If the parting direction is out of the RD of the primitives, $\boldsymbol{d} \notin \boldsymbol{d}_\alpha$, the parting plane should be located at the a plane passing through the axis of the spherical surface and cone frustum, and located on side surfaces for the pyramid frustum. In the later case, the side core should be used.

Table 4.3 Parting position feature in different primitives

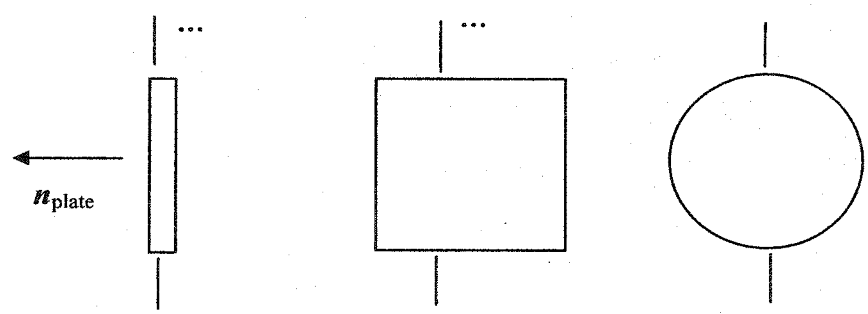
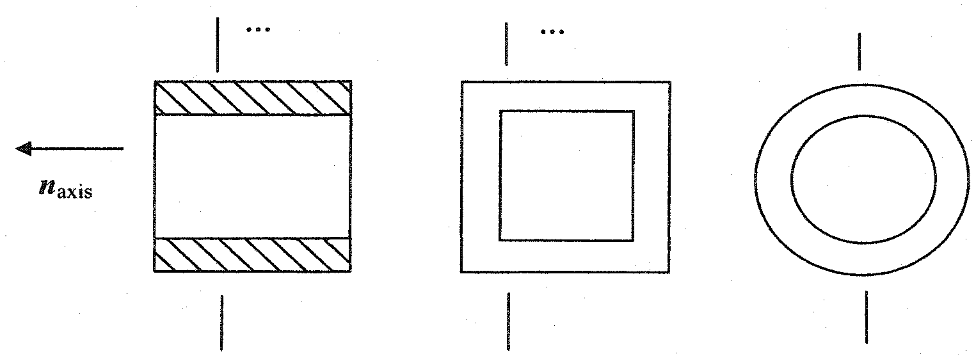
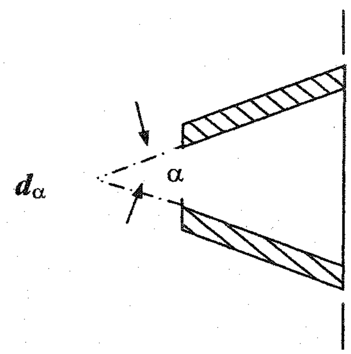
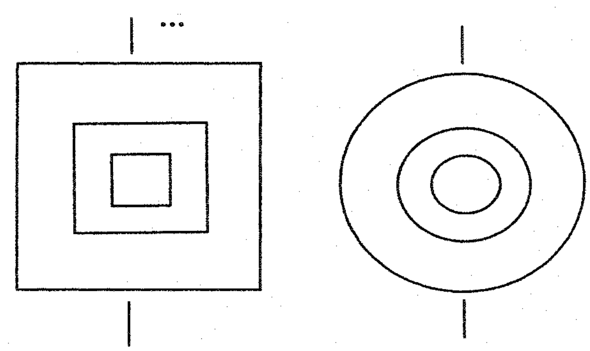
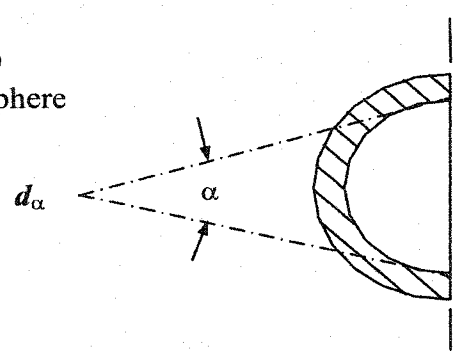
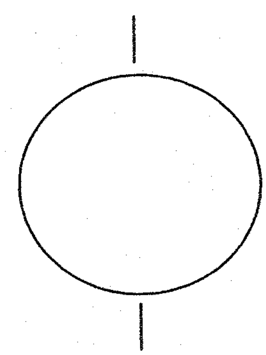
Primitive	Parting plane position
a) Plate	 I: $n_{plate} \bullet d \neq 0$ II: $n_{plate} \bullet d = 0$
b) Sleeve	 I: $n_{axis} \bullet d = 1$ II: $n_{axis} \bullet d \neq 1$

Table 4.3 Parting position feature in different primitives (continued)

Primitive	Parting plane position	
c) Frustum	 I $d \in d_\alpha$	 II $D \notin d_\alpha$
d) Sphere	 I $d \in d_\alpha$	 II $d \notin d_\alpha$

For each primitive, a pair of short solid lines indicates a unique parting position and the line followed by ellipsis indicates that there are many parallel parting positions available in the specific parting direction. The unique parting position can be determined as the periphery of the projected area of the primitive along the parting direction is formed by edge(s), and multiple parting positions are created since the periphery of projected area of the primitive is formed by surface. In the former, the parting plane is placed at the edge(s) of the outer periphery. In the latter case, parting position depends on the specific requirements for parting plane or depends on the experience knowledge.

4.4 Philosophy of parting scheme generation

The parting scheme design is divided into three stages, namely, scheme generation, scheme evaluation and scheme optimization. The task of the scheme generation includes the determination of parting scheme and parting plane. This stage uses the geometrical feature to generate all possible schemes. The geometrical feature includes the CR and RD. The function of the scheme evaluation is to perform a comprehensive evaluation on the generated design schemes. The evaluation is conducted based on the rule-based models. Each scheme is examined by applying a series of evaluation models to different aspects of the parting scheme. The performances of different schemes can thus be obtained. According to the results obtained from the evaluation stage, an optimization of schemes could be performed. The parting scheme with the best performance will be selected as the final scheme based on the optimization algorithm. An optimal or near optimal scheme will be formulated.

The flow chart of the parting scheme design is shown in Fig 4.2. After the casting model is input, the geometrical feature is analyzed in to obtain the CR and to calculate the RD of the given casting. Based on the RD, the match between the candidate parting directions and planes with maximum projected area is performed and the candidate parting schemes are generated. The candidate parting schemes are evaluated according to the evaluation model. Then an optimal scheme can be generated by the optimization procedure.

The process control of parting scheme generation is based on the dependency of the logical relations between each design stage. In the following sections, the generation of parting scheme is described.

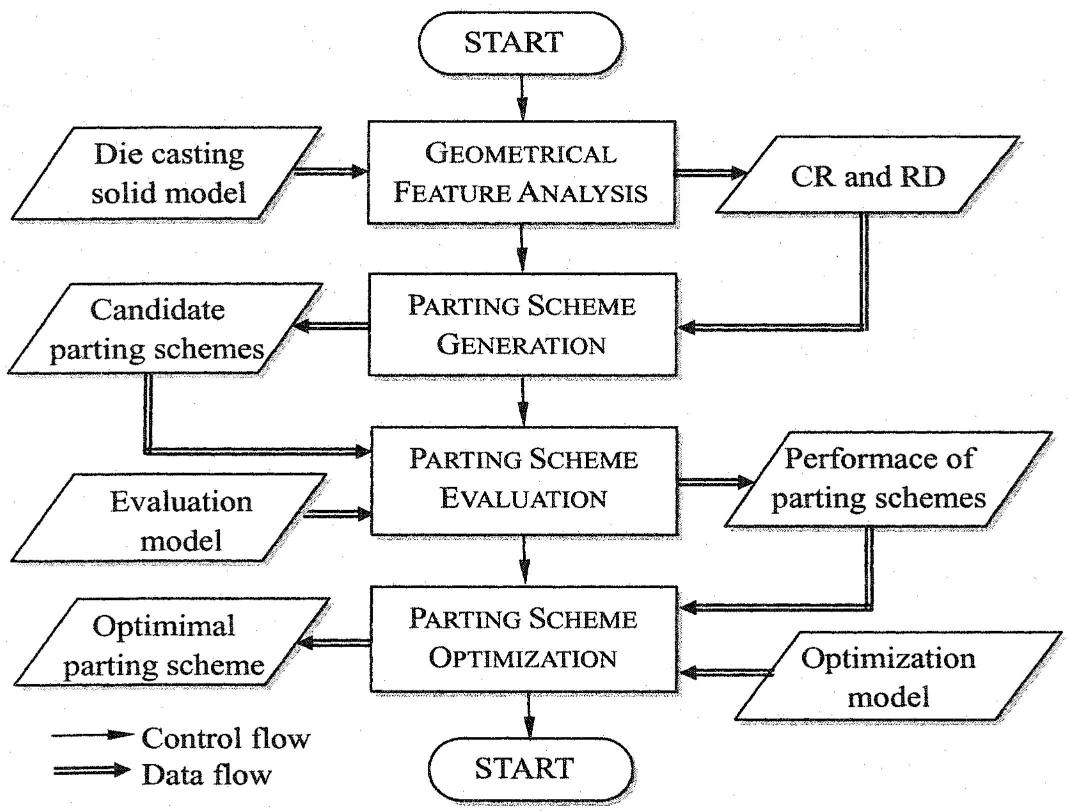


Fig. 4.2 The dependency graph of parting scheme design

4.5 Generation of parting schemes

As previously discussed, the parting scheme design is a problem of "single input/multiple output" in nature. For any given casting object, a number of parting alternatives may exist, in other words, parting scheme may not be unique for a given casting [Nainy et al 1997, Fischer and Ramaseshadri 1997]. Variations in customer's requirement, quality specification, manufacturing process and economical consideration may lead to different solutions for the same casting. An optimal parting scheme is generated from a number of candidate parting schemes by synthesizing various design factors. The purpose of the parting scheme generation is to find all possible schemes for obtaining an optimal one. In order to generate all scheme alternatives, the following two problems should be solved:

i) Determination of design space

ii) Generation mechanism of design scheme

The design space refers to the set of all feasible design solutions [Balachandran 1993]. It determines the range in which the design will be carried out, and involves two aspects. One is the determination of design variables, and the other is the determination of the value range of design variables. Generation mechanism of the design scheme refers to design method and design sequence, i.e., design strategy. The former determines where to design, and the later determines how to design. In parting scheme design, the generation of candidate schemes can be regarded as matching between the candidate parting directions and candidate parting surfaces. The topic involves which candidate parting directions are activated and how the matching between the candidate parting directions and candidate parting planes is performed. In real die design, a common practice is to visualize a number of parting alternatives, then an optimal one is selected from the alternatives based on the design rules and design requirements. Similarly, in a computerized design system, all possible parting schemes should be tried so that the optimal one can be included. In addition, the methodology to generate parting scheme is also important which could generate feasible schemes and avoid unnecessary matching procedures and undesirable schemes. The design space and scheme generation mechanism are discussed in the following section.

4.5.1 Design variables and design space of a parting scheme

As discussed earlier, a parting scheme can be regarded to be a successful match between a particular direction and a particular plane. Therefore, there are two design variables in parting scheme design, i.e., parting direction and parting plane. A parting scheme can be expressed as follows:

$$ps=(cpd, cpp) \tag{4-1}$$

Where *ps* denotes a parting scheme, *cpd* denotes a direction which is selected as parting direction, and *cpp* represents a plane on a casting which is matched as parting plane corresponding to the *cpd*. There are two ways to match between parting direction and parting plane. These are

i) Direction-priority match

ii) Plane-priority match

In the first match, a direction is selected and fixed. The direction is then activated as parting direction and is used to match a proper plane to generate a parting scheme. In the second match, a feasible parting plane is selected and fixed, and the plane is used to match the proper parting direction. In the proposed system, the first way is introduced since the release direction of a cast object can be well obtained. Moreover, this way is in better accord with the actual practice of parting scheme design. Therefore, the design space of parting scheme is dependent on the range of feasible parting directions. The range of feasible parting directions is analyzed and the design space will then be defined.

In principle, any direction can be selected as the parting direction for a given die casting. With considerations stemming from die manufacturing, use of side cores, filling condition and requirement of die casting process etc., in fact, there is only a direction or a set of directions suitable for serving as parting direction. The question arises as to how select the most feasible parting direction. A general rule is that the parting direction should be so selected that the number or volume of side cores is kept to a minimum, i.e., a preferred parting direction is such one along which the minimum number or volume of interference elements (IE) is incurred. From this consideration, a parting direction should be included at least in one of the RD_{CR} in a die casting, i.e., a parting direction should release one CR at least. Therefore, all the feasible parting directions should be included in the set of directions which presents the sum aggregate of all RD_{CR}s, i.e.,

$$CPD = \{RD_{CR1} \cup RD_{CR2} \cup \ldots \cup RD_{CRi} \cup \ldots \cup RD_{CRn}\} \tag{4-2}$$

where n is the total number of CR in a die casting.

The design space of parting scheme can be described as follows: for a given cast object, the set of directions which is included at least in one RD_{CR}, or, the set of directions which can at least release one CR is defined as the design space of parting scheme. The design space of parting scheme is equal to the set CPD, the sum aggregate of all RD_{CR}s in a die casting. The set of CDP is called the set of candidate parting directions. Any direction included in the CPD is called a candidate parting direction, denoted as $cpd \in$CPD. Since the candidate parting directions include all the possible parting directions, the parting schemes generated include all possible schemes, and certainly cover the optimal one. It should be noted that the CPD may be an unlimited directional set.

4.5.2 Generation mechanism of parting schemes

Once the design space of parting scheme has been determined, the design can be carried out and parting schemes can be generated. According to the analysis in the last section, the set of directions, $cpd \in$CPD, is the design space of parting schemes. A parting scheme is a match between a particular direction and a particular plane. The process of parting scheme generation is a match process between specific directions and specific planes. In principle, a candidate parting direction corresponds to a parting scheme. Since the direction-priority match is introduced, all the possible parting schemes can be generated if all directions in CPD are matched with the parting plane alternatively. All the possible parting schemes can be expressed as follows:

$$\text{CPS}=\{ps(cpd_i): cpd_i \in \text{CPD}\} \tag{4-3}$$

Where $ps(cpd_i)$ denotes the parting scheme corresponding to the candidate parting direction cpd_i. The set of CPS is called the set of candidate parting schemes. Any scheme *ps(cpd)* is called a candidate parting scheme. The CPS may be an unlimited set. Since the candidate parting schemes are generated by activation of all candidate parting directions respectively, the parting schemes generated include all possible schemes. The match procedure is repeated until the candidate directions in the CPD are exhausted. From the above discussion, it can be found that the generation of parting schemes includes two basic actions:

i) Alter the candidate parting directions

ii) Match the suitable plane to create parting schemes

In the following sections, the mechanism of activating the candidate parting directions and matching suitable plane is described.

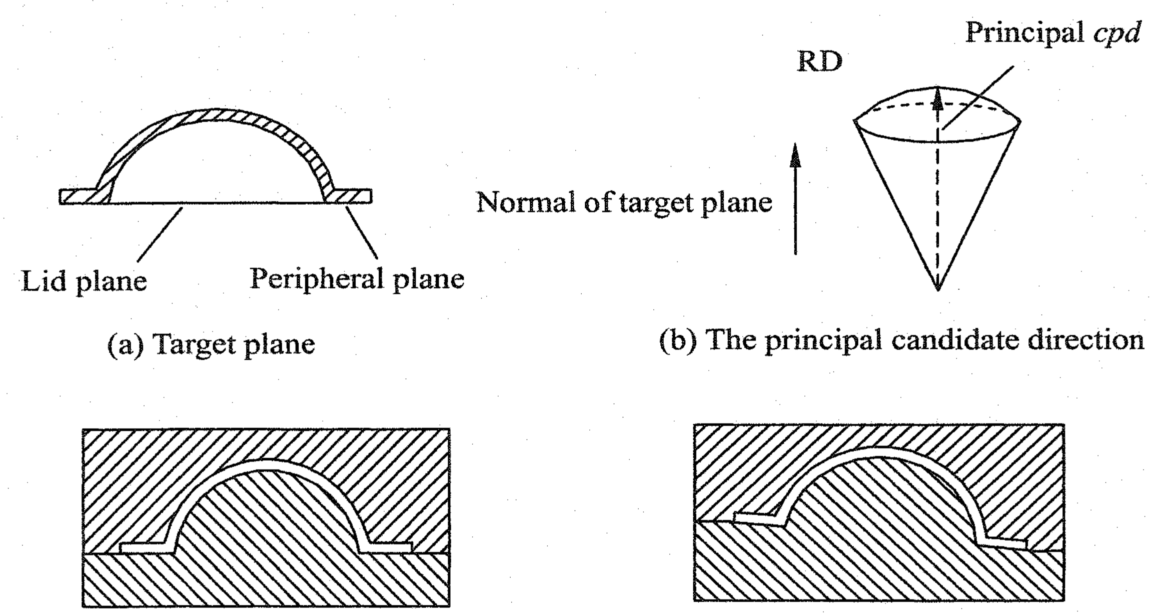
4.5.3 Activation of candidate parting directions

After RD analysis, CPD of a cast object can be obtained. Each direction included in the CPD is qualified as a parting direction. All of the directions in the CPD will be activated in turn to match the corresponding parting surface. The direction which is being activated to match parting surfaces is referred to as an active candidate parting direction. If a lot of candidate directions occur in CPD and every candidate parting direction is used as an active parting direction once, the amount of match operation may be enormous. It is necessary to develop an efficient strategy to identify the most suitable candidate parting directions so as to eliminate unnecessary direction activation and raise the efficiency of candidate parting scheme generation.

In the theory of state space search, a state space may be searched in two ways: from given data of a problem instance toward a goal or from a goal back to the data [George 93]. In data-driven search, or called forward chaining, a problem solver begins with the given facts of the problem. Search proceeds by applying rules or facts to produce new facts. This process continues until it generates a fact that satisfies the goal condition. The goal-driven search focuses in the goal, finds the rules that could produce the goal, and chains backward through successive rules and sub-goals to the given facts of the problem. In this section, two search ways are combined to locate the candidate parting directions. Since there may exist enormous undesired candidate parting directions, these undesired directions should be pruned from the set of candidate parting directions according to certain design rules or design sub-goals. And then a forward chaining activation is performed for a new set of candidate parting directions. Based on this consideration, a principal candidate parting direction is proposed and used as activate parting directions. The principal parting direction could be found as follows.

The principal candidate parting direction (*pcpd*) is proposed for eliminating unnecessary direction activation and match operation. As discussed previously, there are four types of RD images on RD map, i.e., point, line, polygon or circular image. Except for the point image which represents an unique release direction, all other images may present a set of release directions. In fact, there are a number of candidate directions which do not have to be activated in the polygon, circular, and line or bar images on the RD map. The parting schemes generated with such directions are not at optimum state. Thus such directions can be removed out from the set of CPD, and the removal does not influence the design result. The principal candidate direction refers to an optimal direction or a set of optimal parting directions in a specific RD image defined by a line or a polygon or a circle. In order to illustrate the principal candidate parting direction, a target plane concept is introduced. The target plane is a lid plane or peripheral plane adjacent to a CR surface which can be used as parting plane for a given primitive or CR. There must exist a direction in the RD of the CR which coincides or has minimal included angle with the normal vector of the target plane and has the highest value of RDDF. Such a direction is called the principal candidate parting direction. The rules for principal *cpd* can make sure that the flat parting plane or an inclined plane with minimal inclined angle and with minimal number of side cores is used. Fig. 4.3 shows the principal candidate direction in a RD formed by a simple spherical entity. The RD corresponding to a spherical surface is a set of directions. However, except for the principal candidate direction, there are enormous direction points in the RD image, i.e., there are numerous candidate parting directions. All of these directions can lead to a declined parting plane. The die then becomes complicated, as shown in Fig 4.3 (c) and (d). The generated schemes would not be optimal. The principal *cpd* is determined by the normal of target plane. With the principal candidate direction, an optimal scheme is generated in a given direction activation. The number of direction activation and match operation can be largely reduced.

The *pcpd* of a CR is correlated to the normal of target plane. Suppose that the selection of parting direction for a feature obeys regular orientation, then the *pcpd* corresponding to each type of CR can be defined as shown in Table 4.4.



(a) Target plane

(b) The principal candidate direction

(c) Flat parting plane determined by the principal *cpd*

(d) Declined parting plane detemined by any non-principal *cpd*

Fig. 4.3 A principal candidate direction defined by target plane

Table 4.4 The CR and *pcpd*

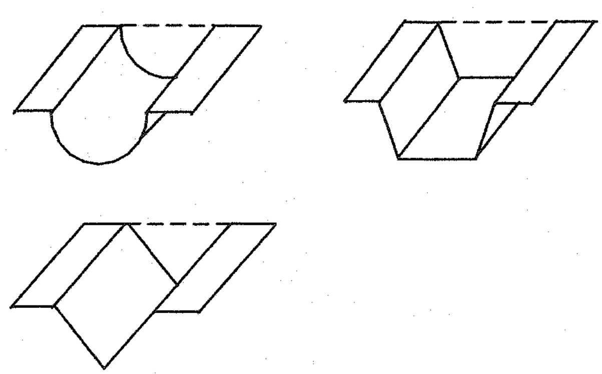
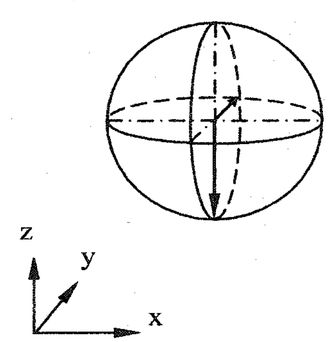
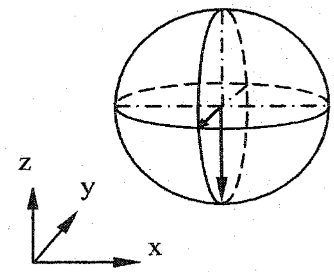
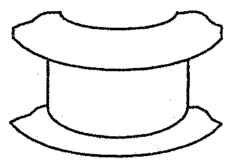
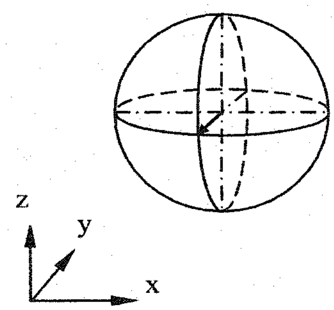
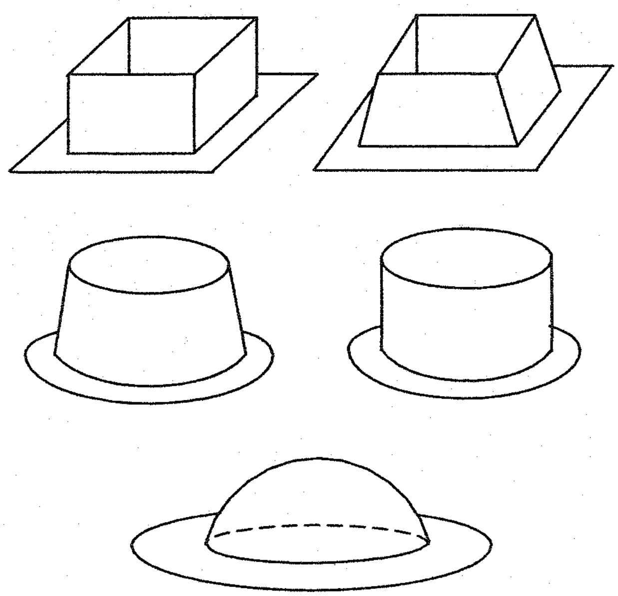
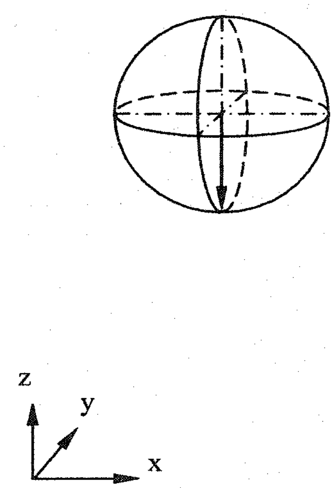
CR	*pcpd*
	

Table 4.4 The CR and *pcpd* (contimued)

CR	*pcpd*
	
	
	

By defining *pcpd*, a large number of candidate parting schemes are pruned, and a new set of directions is created. The PCPD can be expressed as follows:

$$\mathrm{PCPD}=\{\,pcpd_i,\ i=1,\ 2,\ \cdots n\} \tag{4-4}$$

The PCPD is a subset of CPD, i.e., PCPD⊂CPD. In fact, the set of CPD is an unlimited set and PCPD is a limited set. The advantage of introducing *cpcd* is that activation of the candidate parting directions can be completed in a limited number of match procedures and a local optimal parting scheme can be caught in the first activation. Unnecessary match and a great deal of similar schemes can be removed. In addition, numerous sequent evaluation and optimization operations are avoided.

After the set of PCPD has been established, it is only needed to activate all *pcpd* in the PCPD set. Release direction distribution factor (RDDF) indicates the priority of a candidate direction in the consideration of minimal use of side cores. The RDDF can be used to construct the activation sequence of candidate parting directions. In general, the parting direction with minimal use of side core is the preferred parting direction. Therefore, direction activation is conducted from candidate parting directions with high value to that with low value of RDDF. In such a way, the activation of directions is simple and becomes a routing problem. This is a forward chaining activation until all *pcpd*s in PCPD are exhausted.

4.5.4 Selection of parting position

4.5.4.1 Parting position rule

A parting scheme consists of a parting direction and a parting plane. Once a specific parting direction is given, a proper position should be selected as parting plane. The selection of parting position obeys the rule of maximum projection area, that is, the parting position should be located at the cross-section or sections forming the largest projected area. Locating parting plane at the cross-section or sections forming the largest projected area can ensure minimal use of side cores in a given parting direction. This facilitates the removal of the object from a cavity, and the manufacture of the die. The cross-sectional area gradually decreases from the parting surface to points farthest away from the parting surface. This also facilitates cavity-filling and venting.

The parting plane is related to the specific parting direction, the rule of maximum projection area and the relationship between the parting direction and parting plane as shown in Fig. 4.4. When $pcpd_1$ is selected, the cross-section or sections forming the largest projected area is section ⓐ, and the parting position is at plane ⓐ. If $pcpd_2$ is selected, the cross-section or sections forming largest projected area is section ⓑ, and the parting position is at plane ⓑ. If $pcpd_3$ or $pcpd_4$ is selected, the cross-section or sections forming largest projected area is section ©and ⓓ, and the parting position is at plane ©and ⓓ. Different parting direction corresponds to one or a set of parting positions. The position of parting plane on a primitive depends on the geometry and orientation of the primitive and is shown in Table 4.3. Based on the parting feature, parting position of primitives can be determined.

4.5.4.2 Parting position modes

In general, there are four basic modes for locating parting position as shown in Fig. 4.5. The four basic modes are called edge mode, middle protrusion mode, tower mode and

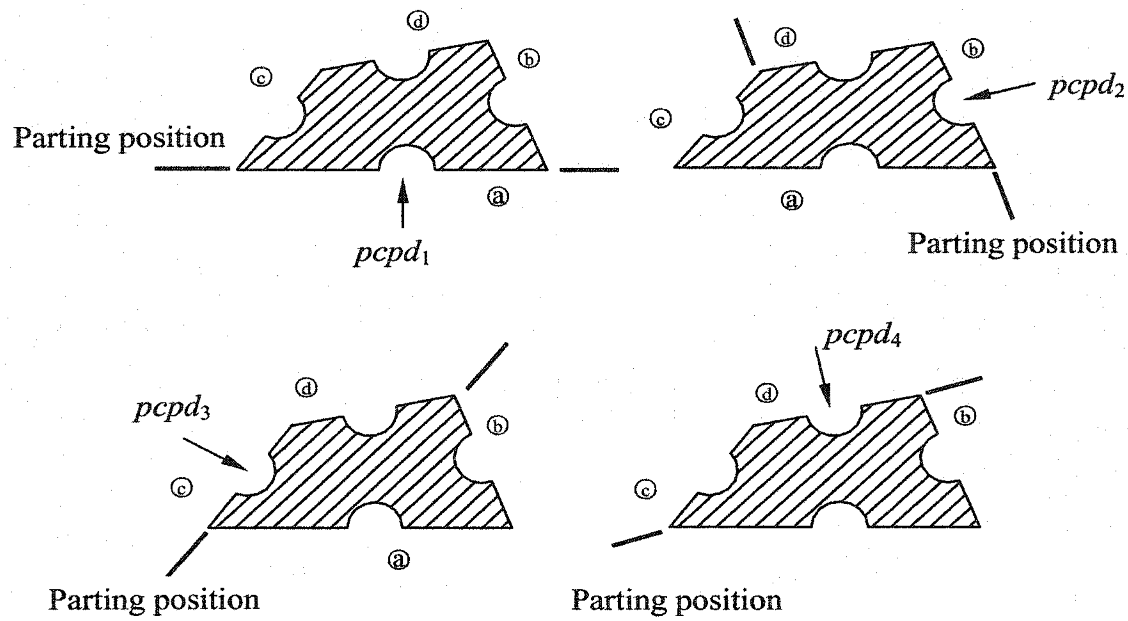
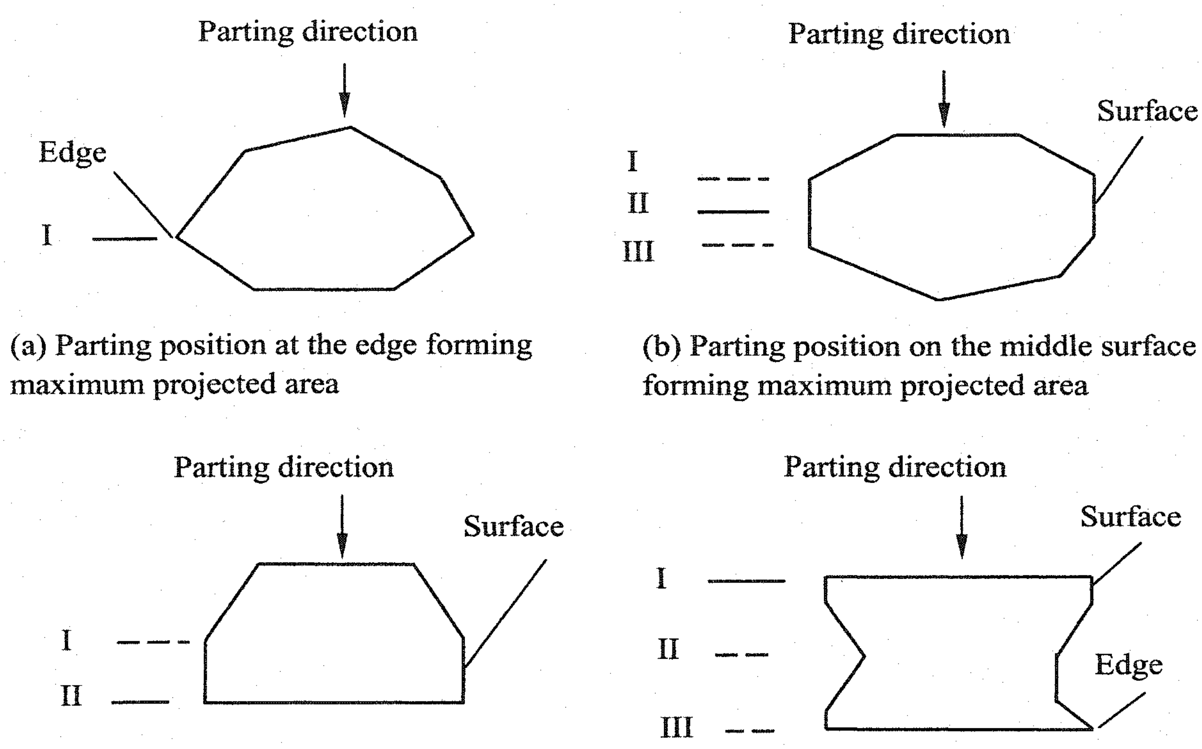


Fig.4.4 Parting position for four parting directions

middle sunken mode respectively. A parting position can be uniquely determined if the boundary of the maximum projected area is formed by an edge or a set of edges on a casting as viewed along the parting direction. This case is shown in Fig. 4.5 (a). In this case, the parting position is at the edge, and the parting position is uniquely determined. If the boundary of the maximum projected area is formed by a vertical surface or a set of vertical surfaces on a casting, the parting position cannot be uniquely determined since the section forming the boundary of the maximum projected area is not unique. This case is shown in Fig.4.5 (b). Extra rules or heuristic rules are required in this case for the determination of the parting position. As the matching strategy, the position II should be firstly recommended. If the parting mark is unacceptable on the vertical surface, the parting position II should be removed, and the parting position can be selected from the parting position I and III. In the third mode shown in Fig.4 5 (c), the end surface is selected as the parting position. For the last mode shown in Fig.4 5 (d), the end plane is selected as parting position. The positions of dashed line denote the optional position of parting plane. In any case, a flat parting plane should be preferred because the flat parting plane can simplify the die structure, and the draw height should keep minimal because small draw height is favorable for filling. In fact, primitives involved in parting position are different. The parting plane can be determined by examining the mode and the orientation of the primitives with the largest projected area.



(a) Parting position at the edge forming maximum projected area

(b) Parting position on the middle surface forming maximum projected area

(c) Parting position at end plane

(d) Parting position at end plane

Fig. 4.5 Parting modes in different features

4.5.5 Matching procedure

After a direction is activated as the parting direction, the matching operation of the parting plane is carried out. Matching follows four steps:

i) Determine the surfaces or edges with maximum projected area of the periphery,

ii) Determine the primitives involved,

iii) Determine the parting mode, and

iv) Detemine the parting position.

At first, the surfaces or edges with maximal projected area should be tested along the parting direction. It is noted that the surfaces or edges are direction-dependent, as shown in the Fig. 4.4. Test operation is performed by sweeping the solid casting. The points on the most outer contour are recorded. Then the points are examined to determine the primitives which the points lie on. If the periphery of the of maximal projected area is the edge of a primitive, the parting position is uniquely determined. If the periphery of the maximal projected area is a surface of a primitive, the parting mode should be examined to determine the parting position. The matching procedure is shown in Fig. 4.6.

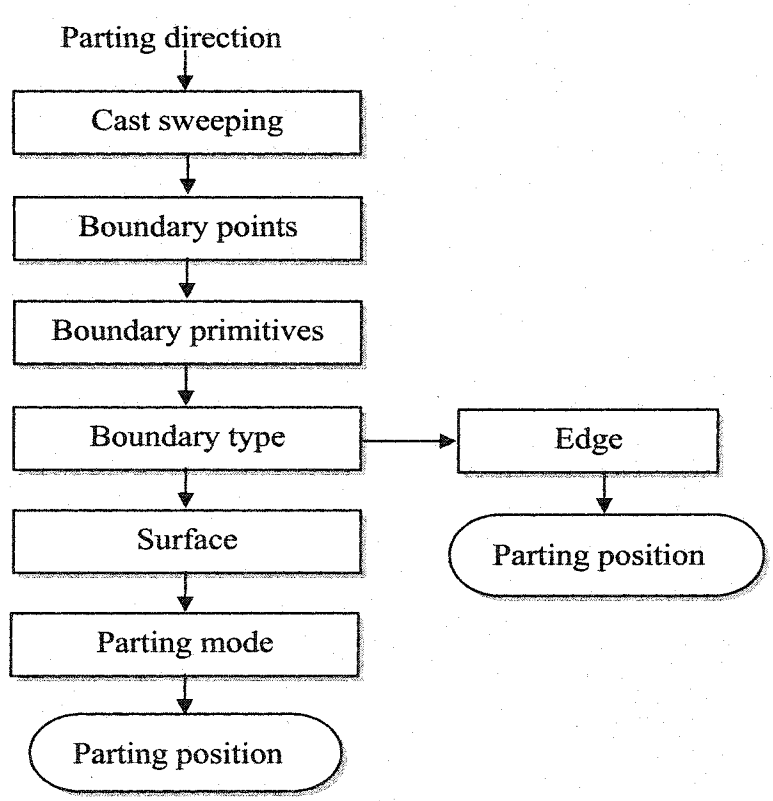


Fig. 4.6 Matching procedure

4.5.6 Design process control

After a set of *pcpd* is created, the control module fires the alteration action. The alteration starts from a *pcpd* with the highest RDDF value, a forward chaining activation is used. When the *pcpd* is exhausted, all the candidate parting schemes are generated, and the evaluation and optimization procedure are followed. The control flow chart of the parting scheme generation is illustrated in Fig. 4.7.

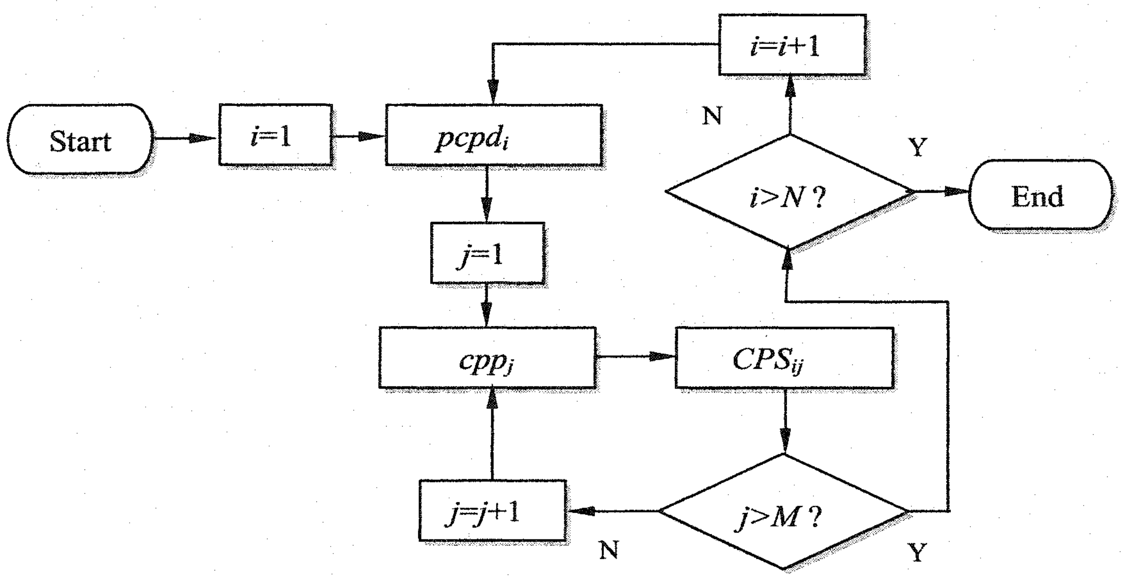


Fig. 4. 7 The control flow chart of the candidate parting schemes generation

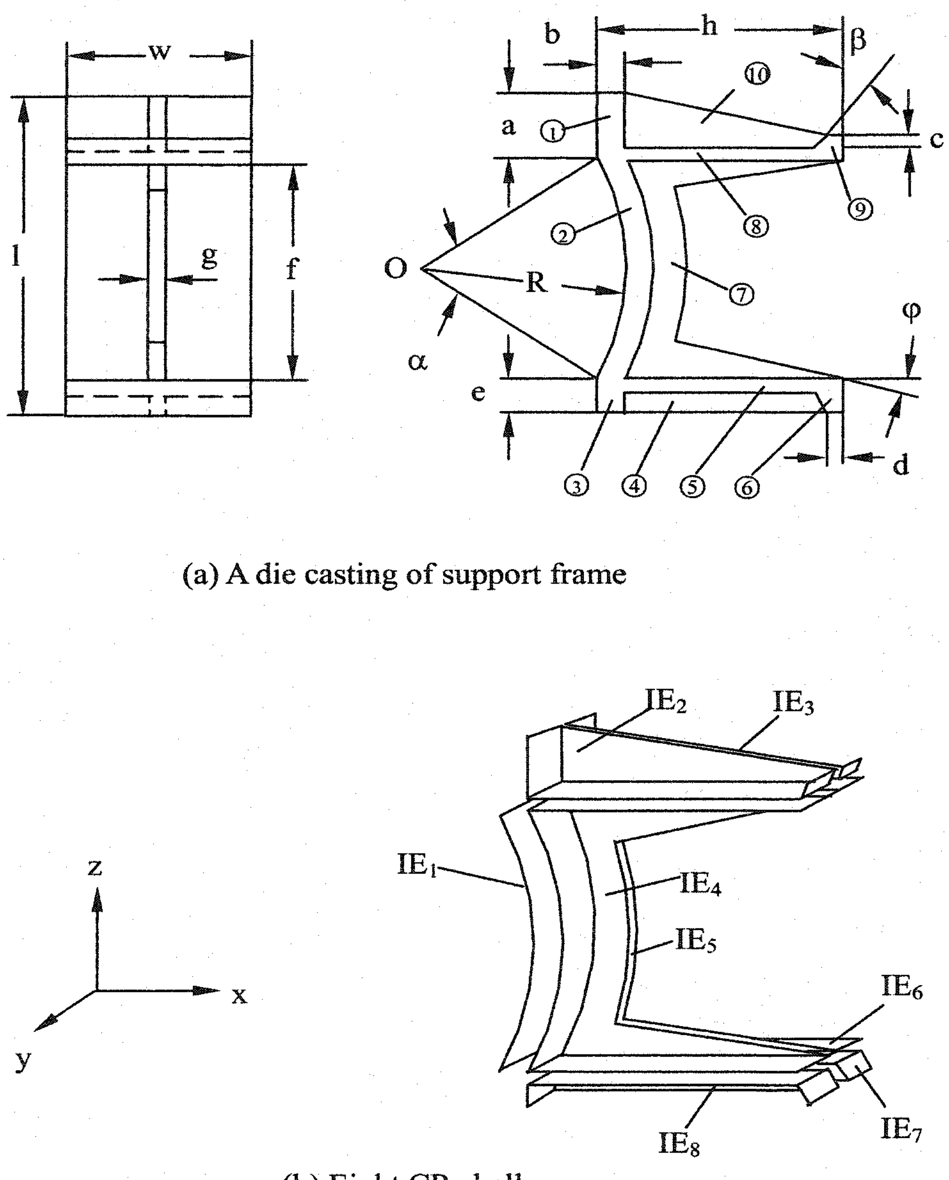
In Fig. 4.7, $pcpd_i$ is ith principal candidate parting direction and cpp_j is $j$the parting position. CPS$_{ij}$ is a candidate parting scheme corresponding to ith principal candidate parting direction and jth parting position. N denotes the total number of *pcpd* in PCPD set, and M denotes the number of *cpp* for a specific *pcpd*. After a *pcpd* is fired as an active parting direction, the matching operation should be done in the active direction. At least a candidate parting scheme is generated in each candidate parting direction. After the match operation is completed in the active *pcpd*, the next candidate direction will be fired and next match will be performed. This procedure is repeated until all the principal candidate parting directions are exhausted and all the candidate parting schemes are generated.

4.6 Design examples

In this section, two design examples are given. The first one is an illustrative example and the other is based on a real die casting.

Example I

The first example is a support frame as shown in Fig.4.8 (a). After the die casting has been modelled, the control module calls the CR recognition routine to extract the CR shells. In the die casting, there are eight CRs, $CR_{1\text{-}8}$. CR_1 is composed of a curved surface. $CR_{4\sim6}$ are composed of a curved surface and three flat surfaces respectively. $CR_{2\sim3}$ and $CR_{7\sim8}$ are composed of four flat surfaces respectively. These CR shells are shown in Fig 4.8 (b). Then the control module calls the RD calculation routine to calculate the RD for each CR and RDDF. According to the CR feature, the principal candidate parting directions of each CR is determined. The *pcpd* set is then created. The RD and *pcpd* of each CR are shown in Table 4.5. Angles indicated in RD are dihedral angles shown in Fig.4.8. The candidate parting directions are the sum aggregate of all RDs, CPD=$\{cpd\text{: } cpd \in RD_{CR1} \cup \cdots \cup RD_{CR8}\}$. The set of CPD is shown in Fig.4.9(a). It can be seen that the RD images are partial spherical surface except RD_{CR6} and RD_{CR4} (opposite RD_{CR6}) which are line images. Any direction within the aggregate can release at least one CR. Fig. 4.9 (b) shows different regions of the RD distribution. The RDDF for each *cpd* can be obtained by counting the number of times each point on the unit sphere is covered by the RD images. The values of DFRD of each region are given in Table 4.6. The value of RDDF indicates the relationship between the number of side core and parting direction. A CR can be released if the parting direction is included in the RD_{CR}, otherwise the CR is an interference element.



(a) A die casting of support frame

(b) Eight CR shells

Fig. 4.8 A die casting and its CR shells

Table 4.5 Feature and RD of each CR

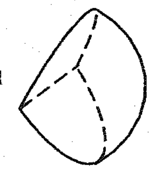
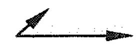
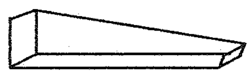
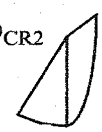
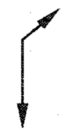
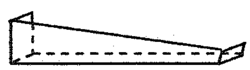
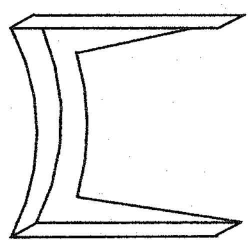
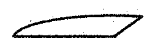
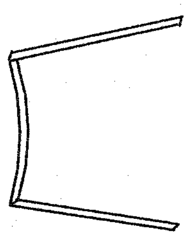
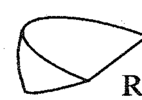
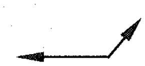
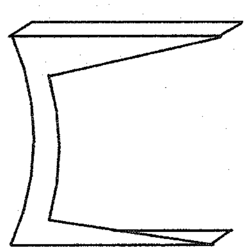
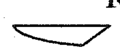
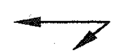
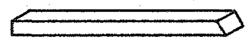
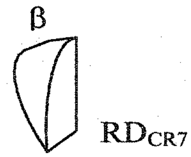
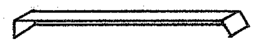
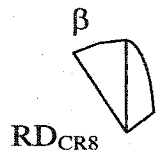
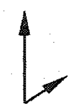
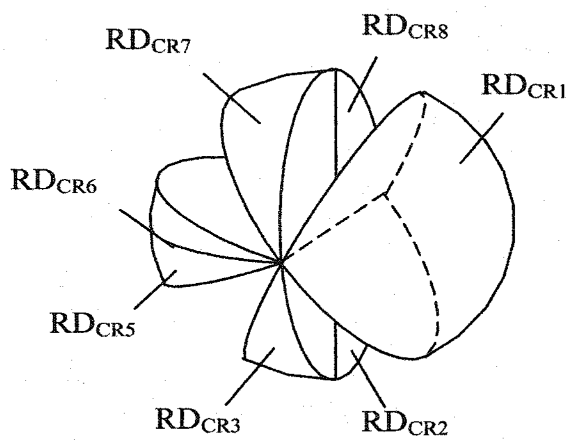
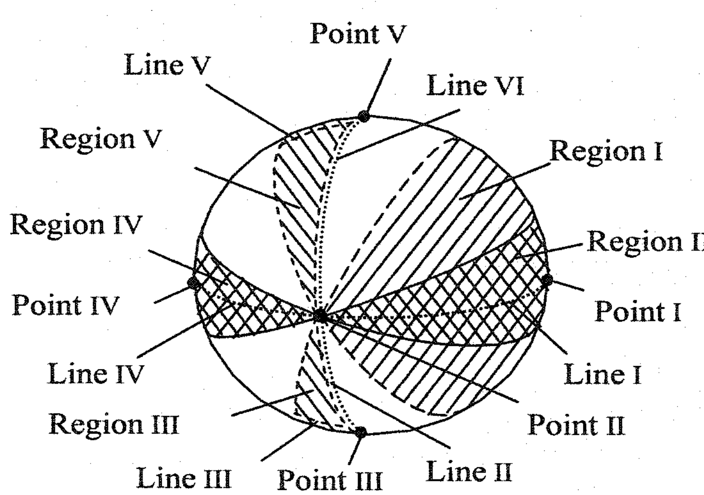
	CR shell	Feature	RD	Principal *cpd*
CR_1		I (1) Through groove	D_{CR1}  π-α	
CR_2		II (2) Blind groove	RD_{CR2}  β	
CR_3		II (2) Blind groove	 RD_{CR3} β	
CR_4		II (2) Blind groove	 RD_{CR4}	
CR_5		I (1) Through groove	2φ  RD_{CR5}	
CR_6		II (2) Blind groove	RD_{CR6} 	

Table 4.5 Feature and RD of each CR (continued)

	CR shell	Feature	RD	Principal *cpd*
CR_7		II (2) Blind groove	 β RD_{CR7}	
CR_8		II (2) Blind groove	 β RD_{CR8}	



(a) The candidate parting directions



(b) RD distribution

Fig.4 9 The candidate parting directions and RD distribution

Table 4.6 RDDF value of each region

Regions	Covered by	RDDF
Point I (Point IV)	RD_{CR1}, RD_{CR4}, RD_{CR5}, RD_{CR6}	4
Point II	RD_{CR1}, RD_{CR2}, RD_{CR3}, RD_{CR4}, RD_{CR5}, RD_{CR6}, RD_{CR7},RD_{CR8}	8
Point III (Point V)	RD_{CR2}, RD_{CR3}, RD_{CR7}, RD_{CR8}	4
Line I	RD_{CR1}, RD_{CR4}, RD_{CR5}	3
Line II	RD_{CR3}, RD_{CR8}	2
Line III	RD_{CR2}, RD_{CR3}	2
Line IV	RD_{CR1}, RD_{CR5}, RD_{CR6}	3
Line V	RD_{CR7}, RD_{CR8}	2
Line VI	RD_{CR2}, RD_{CR7}	2
Region I	RD_{CR1}	1
Region II (Region IV)	RD_{CR1}, RD_{CR5}	2
Region III	RD_{CR3}	1
Region V	RD_{CR7}	1

According to the *pcpd* of each CR listed in Table 4.4, the PCPD set can be created, i.e., PCPD=$\{pcpd_1, pcpd_2, pcpd_3\}$, and the principal *cpd* map can be built as shown in Fig. 4.10. In the given cast part, $pcpd_1$, $pcpd_2$, and $pcpd_3$ just coincides with three coordinate directions respectively. The points on the RD map represent the principal *cpds*. Once the principal *cpds* have been determined, the next step is to activate the directions to generate the candidate parting schemes. There are three pairs of principal *cpds* for the casting, and the three pairs of *pcpds* are activated in turn to match the parting position respectively.

According to the priority of RDDF, the $pcpd_1$ represented by the point II is firstly activated since it has the highest RDDF value. From the discussion earlier, the parting position is

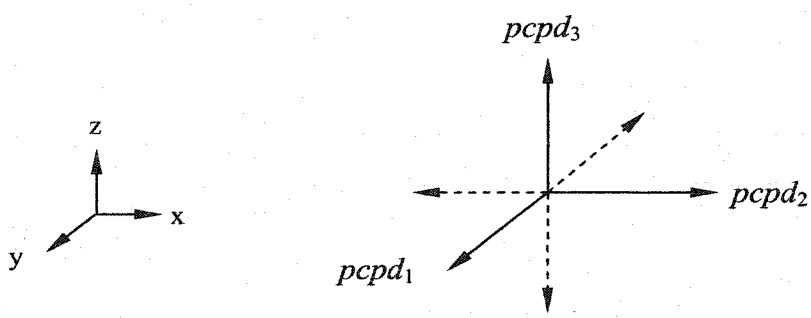
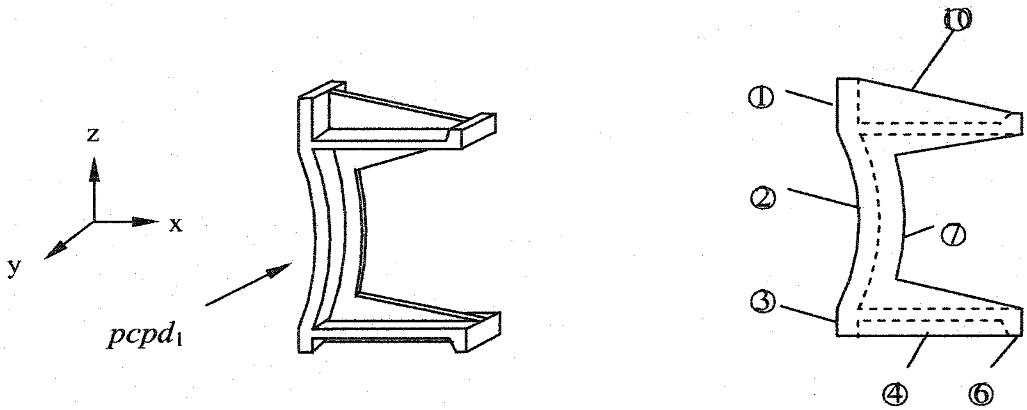


Fig. 4.10 Three pairs of *pcpd*s

located at the surface(s) or edge(s) forming the maximum periphery of the projected area as viewed along the parting direction. After the active direction is activated, the system sweeps the mesh domain to determine the surface(s) or edge(s) forming the maximum periphery and the primitives involved in the periphery. In $pcpd_1$, the maximum periphery of the projected area is formed by the primitives ①, ②, ③, ④, ⑥, ⑦, ⑨ and ⑩, as shown in Fig 4.11. These include seven plate primitives and a cylindrical primitive. According to the parting feature listed in Table 4.3, the parting position cannot be uniquely determined since the periphery is formed by the surfaces of plate primitives. The matching operation will examine the orientation and position of each primitive, and determine the mode of parting position. Since the normal of the primitives ④, ⑦ and ⑩ coincides with parting direction, i.e., $\boldsymbol{n}_{\text{plate}} \bullet \boldsymbol{d} = 1$, the parting position should lie at the lateral surfaces of these primitives. A level test for the three plate primitives is made to determine whether a flat parting plane, a inclined parting plane, or a step parting plane is generated. If these primitives are at the same level, a flat parting plane can be selected. According to the parting modes, the parting position can be determined. The parting structure is shown in the Fig.4.12 and the case accords with the parting mode B as show in Fig. 4.5. Therefore, a symmetric parting position across the maximal section of the casting can be proposed. The scheme is shown in Fig.4.13. Since the

RDDF of $pcpd_1$ equals eight, all CRs can be released along the direction. Both schemes do not require any side core.



(a) Activation of $pcpd_1$ (b) Surfaces forming maximal projected area

Fig. 4.11 Surfaces with maximal projected area in $pcpd_1$

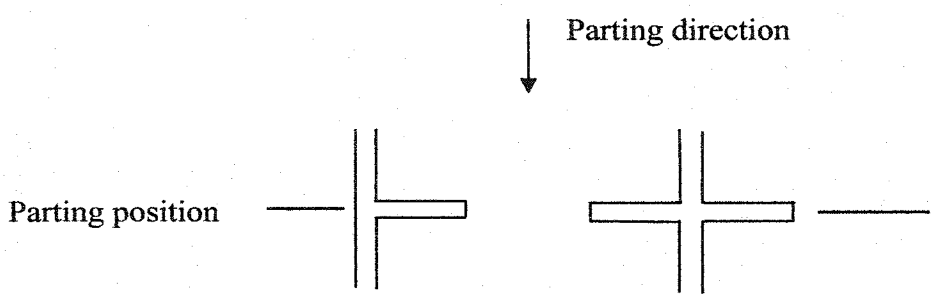


Fig. 4.12 Parting feature

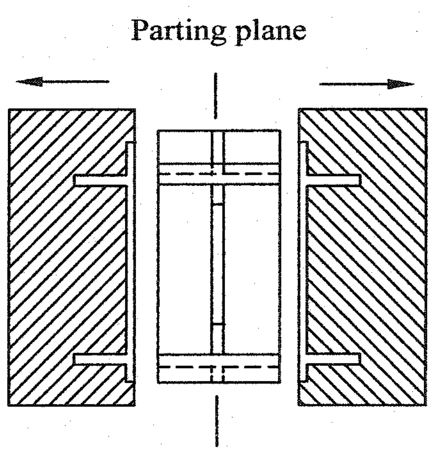
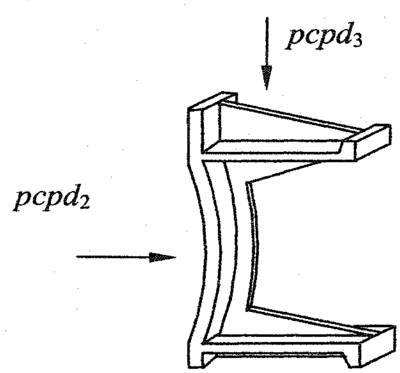
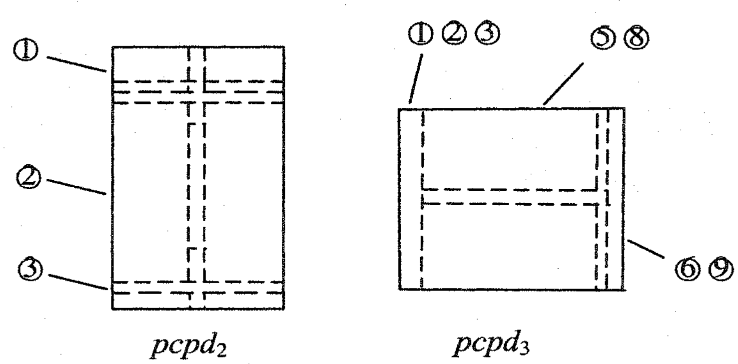


Fig. 4. 13 Parting scheme for $pcpd_1$

In $pcpd_2$ and $pcpd_3$, the corresponding periphery of the maximal projected area is shown in Fig. 4.14. In both cases, the periphery of the maximal projected area is also formed by surfaces, and the parting position cannot be uniquely determined. The side surfaces of primitives ①, ② or ③ is composed of periphery for $pcpd_2$. The system can recognize the parting mode, and the case accords with the parting mode C. Then a parting position across the maximal section of the casting is proposed. The scheme is shown in Fig.4.15. RDDF in $pcpd_2$ is equal to four. There are four CRs which are interference elements and require side slide core to release the casting.



(a) Activation of $pcpd_1$ and $pcpd_3$



(b) Surfaces forming maximal projected area

Fig. 4.14 Maximal projected area in $pcpd_2$ and $pcpd_3$

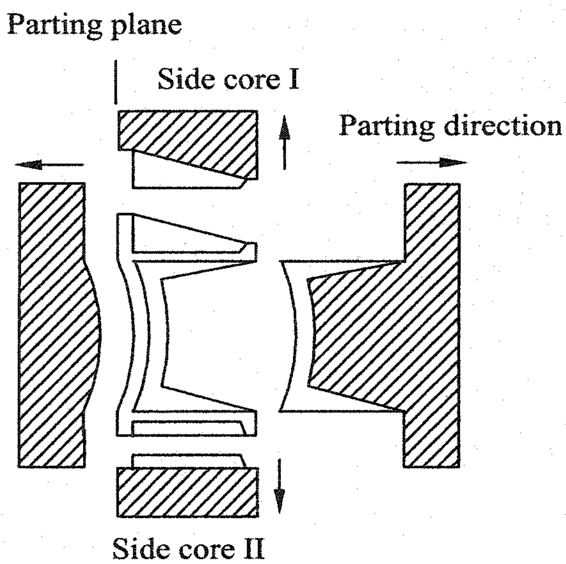


Fig. 4.15 Parting scheme for $pcpd_2$

In $pcpd_3$, the periphery of the maximal projected area is formed by the side surfaces of primitives ①, ②, ③, ⑤, ⑥, ⑧ and ⑨, The case is mode D. A middle parting position is proposed. The parting position proposed is shown in Fig. 4.16. Similarly, there are four CRs to require side slide core to release them since the value of the RDDF equals four.

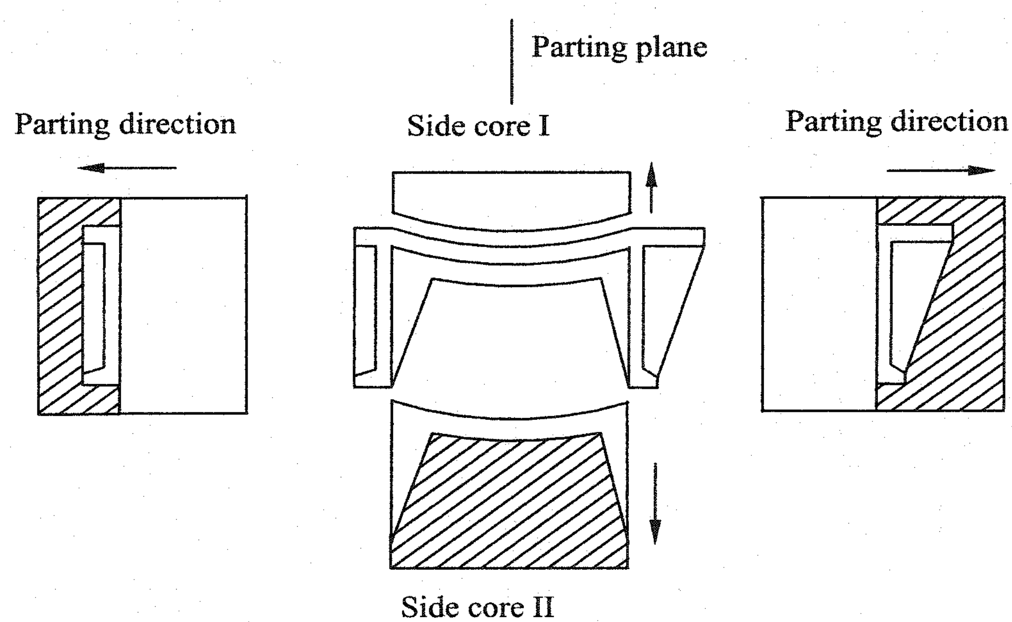


Fig. 4.16 Parting scheme for $pcpd_3$

After each of the principal candidate parting directions are matched respectively, all candidate parting schemes are generated. There are three candidate parting schemes generated in this example.

Example II

To illustrate the parting scheme design for the sleeve category of die castings, a casing body is selected as shown in Fig.4.17. This casing body is a real casting and is employed as an example. The casting consists of a large ellipse sleeve base and a half ellipical sleeve primitive. In the die casting, there are three independent CRs, $CR_{1\text{-}3}$. CR_1 is formed by sleeve inner surfaces, CR_2 is formed by a spherical surface, and CR_3 by a cylindrical surface. The CR shells, RD and *pcrd* of each CR are shown in Fig.4.18. It should be noted that there is an additional principal direction for the sleeve category of die castings. The additional principal candidate parting direction is perpendicular to the maximum plane through the sleeve axis. With two principal candidate parting directions, an unique parting scheme can be generated in each direction since the boundary of the maximum projected area is formed by edges as viewed along the parting direction. The two candidate parting schemes are shown in Fig. 4.19. Parting feature is stored in a frame structure with the parting primitive, topological relationship and parting position in each of the parting primitive. For example, the parting feature is stored in following format for the first parting case.

Parting_primitive: (vertical_ellipse_sleeve)

Parting_position: (ellipse_sleeve_bottom_face)

The parting feature will be transferred to the gating design phase for the configuration of the gates.

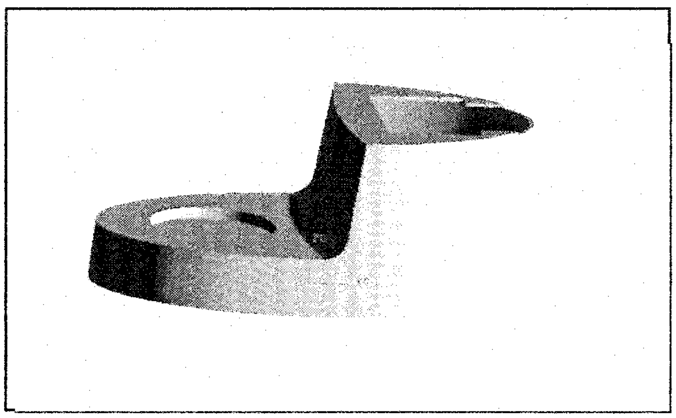
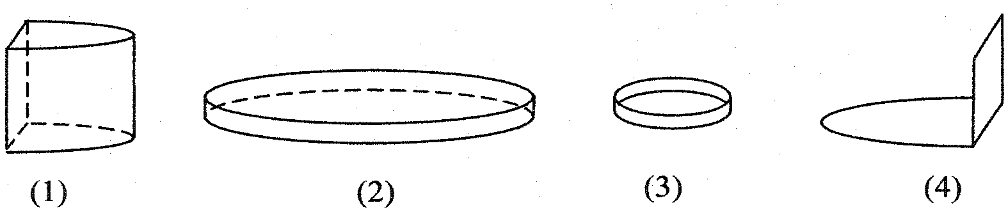
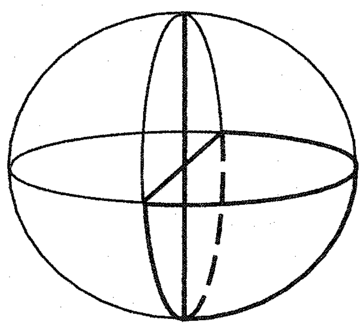


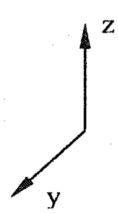
Fig.4.17 Die casting model of a casing body



(a) CR shells

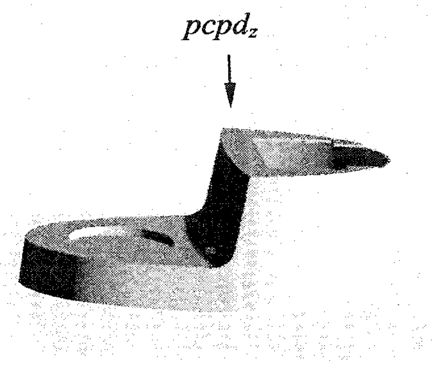


(b) RD Map

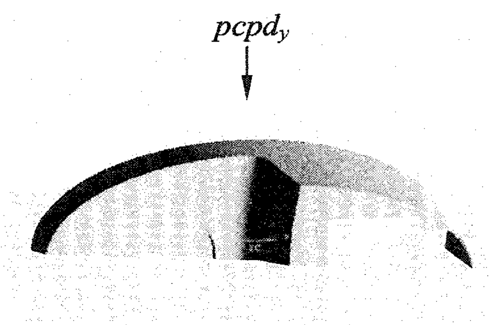


(c) Two *pcpd*s

Fig.4.18 Complex CR shells, RD and *pcrd*



(a) Bottom parting plane



(b) Symmetric parting plane

Fig. 4.19 The two candidate parting schemes

Once all the candidate parting schemes have been generated, evaluation and optimization of the schemes will be required so as to determine the scheme performance and obtain an optimal scheme. The evaluation and optimization of parting scheme is discussed in the next chapter.

4.7 Summary

In this chapter, the following works are revealed:

i) The approach of constructive solid geometry is introduced to model solid casting.

Four categories of geometrical primitives are defined. By prescribing the topological relationship between the primitives, a solid casting can be constructed.

ii) Fundamental features for parting scheme design are discussed. The four groups of ICR features and three groups of RCR features, and four groups of parting features are defined.

iii) The design strategy of parting scheme is described. The principal candidate parting direction (*pcpd*) is proposed to eliminate unnecessary direction activation and matching operation. The selection of parting position obeys the rule of maximum projection area and four basic modes for locating parting position are presented.

iv) Two design examples are given. The procedure and evolution of the scheme generation are illustrated.

CHAPTER 5

EVALUATION AND OPTIMIZATION OF A PARTING SCHEME

5.1 Brief description

After the candidate parting schemes are generated, the question arises as to how a given scheme can be evaluated. Before an optimal scheme is determined, the performance of parting schemes should be understood. This involves evaluation of the parting scheme. The design of a parting scheme is guided by a number of design rules [Edit 1981, Ravi and Srinivasan 1990, Hill et al 1991, Guleyupoglu and Hill 1995]. In practice, designers firstly analyze the geometrical feature of a die casting to be cast, and then apply the design rules to the design. They always try to satisfy all the design rules as possible. However it is often difficult to achieve absolute satisfaction to all rules in a particular design. In general, each rule is considered to represent an aspect of parting scheme performance. Satisfaction of design rule reflects the performance of a design scheme. Therefore, design rules can be used to evaluate parting schemes. In this chapter, parting design rules, evaluation models, optimization and examples of evaluation are described.

5.2 Parting scheme design rules

Over the years, the expertise and experience to determine the parting scheme are well summarized as design rules [Edit 1981, Ravi and Srinivasan 1990, Hill et al 1991, Guleyupoglu and Hill 1995]. In this study, a rule-based approach is introduced for the evaluation of parting scheme. The following lists the design rules commonly used in the design of parting scheme and their interpretation.

Rule 1

The parting plane should be located at the cross-section or sections with largest projected area.

Locating a parting plane at the cross-section or sections with largest projected area can ensure minimal use of side cores in a given parting direction, facilitate removal of the object from a cavity, and facilitate manufacture of the die. The cross-sectional area should gradually decrease from the parting surface to points farthest from the parting surface to facilitate cavity-filling and venting.

Rule 2

The number of side cores used should be a minimum.

The use of side cores increases manufacturing costs, complicates the operation of dies, and slows down the process, it is desirable to reduce the number of side cores whenever possible. Based on this criterion, it can be deduced that the parting direction should be such that the number of CRs to be released by fixed cores is maximal.

Rule 3

The volume of fixed cores should be a maximum.

This criterion indicates the large cores should be incorporated into ejector die or cover die as fixed cores, Small size cores are preferred to be side cores. In such a way, a small and simple withdrawing unit can be used.

Rule 4

A flat parting surface is preferred.

Obviously, the purpose of this criterion is to simplify the die structure.

Rule 5

The cast component moves away from cover die and remains in the ejector die when the die opens.

In order to eject the casting from the cavity, the casting should remain in the ejector die without the aid of any extra mechanism. Any extra mechanism to release the casting would complicate the die and the casting operation.

Rule 6

The height of casting in the parting direction should be a minimum.

The casting height refers to the maximum height of a die casting on both sides of the parting plane along the parting direction. Excessive casting height can cause filling and venting problems. Besides, a deep cavity is difficult to manufacture, and increases the draw distance and the draft of the casting. This criterion facilitates the manufacture of the die and is advantageous to cavity-filling and venting.

Rule 7

The parting surface should not coincide with the datum plane of the cast object.

Flash or parting mark may affect the accuracy of the datum plane. The parting plane should generally not be the same as the datum plane.

Rule 8

The parting plane should coincide with the surface of the casting that requires machining.

Flash or parting mark can be removed during subsequent machining of the casting in one operation.

Rule 9

A particular casting section with requirements of high dimensional accuracy and stability should remain entirely on the same half of the die.

Dimensional accuracy and stability can be better preserved if the section requiring high dimensional accuracy and stability is placed on the same half of the die.

Rule 10

The parting surface should not be allowed to traverse a casting surface on which the parting line mark is unacceptable.

The parting line mark influences the finish of casting surface. Keeping a surface entirely in the same half of die can avoid a parting mark on the surface.

The design rules listed above are important requirements for a parting scheme. However, it is difficult to satisfy all requirements at the same time. With the above rules, an evaluation scheme is developed in the following section to evaluate the degree to which all the design requirements are met for a particular parting line design.

5.3 Modelling of the design rules

Based on the design rules listed in the above sections, an evaluation model is developed. The design rules can be divided into two classifications, namely, design constraints and design criteria. Design constraints stem from the die casting specifications, such as machining surface, datum surface specification and so on. In a design, a constraint is either satisfied or violated, and the evaluation points to either are of the scenario. Design criteria are based on the die structure and cavity-filling, such as minimal use of side cores, minimal draw height and favourable metal filling etc. A criterion may be fully satisfied, or partially satisfied, or dissatisfied at all, and evaluation is based on some form of rating, i.e., the degree of satisfaction. The degree of satisfaction shows to what extent a scheme satisfies each of the criteria.

5.3.1 Model of design criteria

As discussed above, each criterion can represent an aspect of scheme performance. It should be noted that in practice the design criteria used in a parting scheme design are often not expressed as numerical information. Therefore, quantification of the design criteria into computable functions is necessary. Suppose there are n criteria used in the scheme design, and $Sat_j(\text{critr}_i)$ is a satisfaction function, the overall scheme performance of the candidate parting scheme can be expressed as follows:

$$\xi(\text{CPS}_j)=f(Sat_j(\text{critr}_1), \cdots, Sat_j(\text{critr}_i), \cdots, Sat_j(\text{critr}_n)) \tag{5-1}$$

where CPS_j denotes the jth candidate parting scheme, $\xi(\text{CPS}_j)$ expresses the overall scheme performance of the jth candidate parting scheme. $Sat_j(\text{critr}_i)$ denotes the satisfactory level of the *j*th parting scheme to meet the *i*th criterion. It should be noted that the $Sat_j(\text{critr}_i)$ is numerical value. The value will be normalized to the range of 0 to 1. A $Sat(\text{critr}_i)$ value of 0 indicates a dissatisfaction of *i*th criterion, whereas a value of 1 indicates full satisfaction of *i*th criterion, and the intermediate values are the cases between the two extremes. Thus a scheme alternative can be evaluated by a number of function values. By modelling design criteria and defining the satisfactory level, a quantitative evaluation on the parting schemes can be realized. The performance functions for each design criteria are given below.

Function 1 *Satisfactory function for the number of fixed core/side core*

In a die casting, there may exist many CRs, which can be released either by fixed cores incorporated into one of the two die halves or by side slide cores. In order to measure the use of side cores, the ratio of the number of CR which can be released by fixed cores to the total number of CRs is introduced as a satisfaction function on the use of cores:

$$Sat(\text{num\_sc}) = \frac{n}{N} \tag{5-2}$$

where N is the total number of CRs and n is the number of CR released by the fixed core incorporated in the cover die or the injector die. The difference (N-n) represents the number of side cores required. The greater the value of *Sat*(num_sc) is, the less the use of side cores is.

Function 2 *Satisfactory function for volume of fixed core/side core*

Similarly, the CR may be different in volume. The CR with a large volume should be released by fixed core, and the core-withdrawing device has to be simple. The volume function of fixed core/side core is modelled as follows:

$$Sat(\mathrm{vol\_sc}) = \sum_{i=1}^{n} v_i \Big/ \sum_{j=1}^{N} v_j \tag{5-3}$$

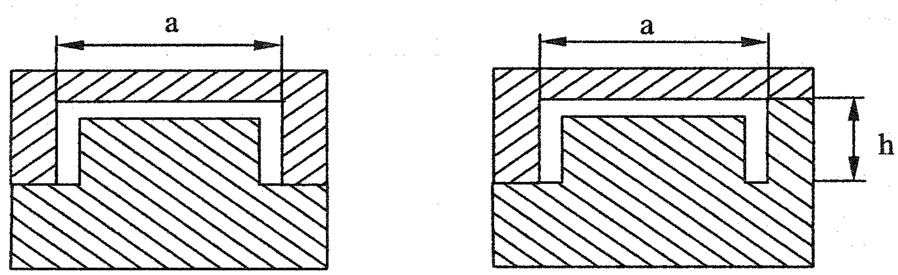
Where v_i is the volume of an independent CR_i, N is the total number of CRs and n is the number of CRs released directly by the fixed cores. $\sum_{j=1}^{N} v_j$ denotes the total volume of CRs and $\sum_{i=1}^{n} v_i$ denotes the volume of CRs released by fixed cores. In the evaluation of a parting scheme, *Sat*(num_sc) and *Sat*(vol_sc) will be ignored if N is equal to zero.

Function 3 *Satisfactory function for a parting plane*

Obviously, a simple parting surface can simplify the die structure. A flat parting plane is the simplest design. The characteristic of a flat parting plane is that the entire parting plane lies at a plane perpendicular to the parting direction. The area of a flat parting plane is identical to the projected area itself. For a complex parting plane, its actual area is greater than its projected area. It is believed that the deference between the actual area and the projected area can indicate the degree of complexity of a parting plane. Therefore the following function can be used to calculate the satisfaction degree of a parting plane:

$$Sat(\mathrm{ps}) = \frac{A_{Pro\_ps}}{A_{ps}} \tag{5-4}$$

where A_{Pro_ps} denotes the projected area, and A_{ps} the actual area of a parting plane. *Sat*(ps) equals unity for a flat parting plane. The smaller the value of *Sat*(ps) is, the more complex the parting plane will be. An example for the calculation of *Sat*(ps) is given in Fig. 5.1. Suppose that parting plane is unit width, the *Sat*(ps) of two parting schemes are given in Table 5.1. In the two parting schemes, the flat parting plane with high *Sat*(ps) is preferred and stepped one with low *Sat*(ps) is next preferred.



(a) Flat parting plane (b) Stepped parting plane

Fig. 5.1 Example for calculating *Sat*(ps) of parting planes

Table 5.1 *Sat*(ps) of four typical parting planes

Parting plane	A_{ps}	A_{pro_ps}	Sat(ps)
Flat	a	a	a/a
Step	a+h	a	a/(a+h)

Function 4 *Satisfactory function for casing height*

This function is used to test the ratio of the vertical size and the horizontal size. The casting height in a cavity is an important factor for the determination of a parting scheme. The allowed casting height is related with the area of parting plane. Therefore, the function for casting height is as follows:

$$Sat(\text{cast\_h}) = 1 - \frac{h_{\max}}{h_{\max} + H_{\max}} \tag{5-5}$$

where H_{max} is the average of the two maximum dimensions of the die cavity in vertical and horizontal directions, and h_{max} is the maximum casting height on both sides of the parting plane. The H_{max} can be calculated by the following formula:

$$H_{max}=(H_{max\_x}+H_{max\_y})/2 \tag{5-6}$$

where H_{max_x} is the maximum size of the cavity in horizontal direction, H_{max_y} is the maximum size of the cavity in vertical direction. The h_{max} is determined by picking the largest one between h_{cast_cov} and h_{cast_eje}:

$$h_{\max} = \max(h_{cast\_\text{cov}}, h_{cast\_eje}) \tag{5-7}$$

where h_{cast_cov} is the height of the casting in the cover die, and h_{cast_eje} is the height of the casting in the ejector die.

Suppose that there are two candidate parting schemes for the example as shown in Fig.5.2, the evaluation results of *Sat*(cast_h) are given in Table 5.2. From the results, the scheme II has considerable higher priority in respect to casting height. The thickness of the casting is assumed to be unity.

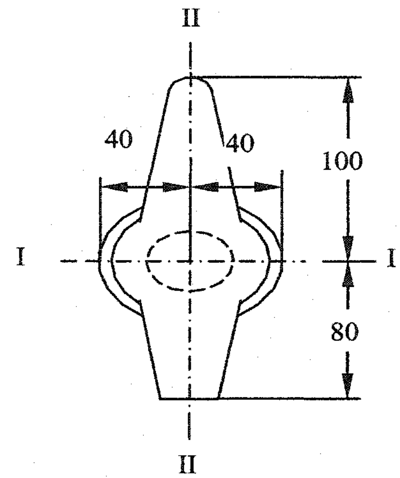


Fig. 5.2 An example for evaluation of casting height

Table5.2 *Sat*(*cast_h*) for two typical schemes

Scheme	H_{max}	h_{max}	Sat(cast_h)
Scheme I	80	100	0.444
Scheme II	180	40	0.818

Function 5 Satisfactory function for holding force

After solidification, the die casting should remain in the ejector die so as to ease its removal. Generally, this is realized by the holding force between the casting and the fixed cores due to the shrinkage of the solidified casting. There are several models to calculate the holding force [Chen 1978, Edit 1981, Han 1985, Wu 1987]. A simple formula used to calculate the holding force is as follows:

$$F_{core}=lhp(\mu\cos\alpha-\sin\alpha) \tag{5-8}$$

where F_{core} denotes holding force on a core; l is perimeter of cross-section of the core, h is height of the core, p is the shrinkage stress, e.g., p for aluminum alloy is 100-120 kg/cm^2

(10-12Mpa). μ is frictional factor between die casting and core, and α is the draft of the core. The parameters used are shown in Fig 5.3. If F_1 and F_2 denote the holding forces on the two die halves respectively, then

$$F_1 = \sum_{half1} F_{core}, \qquad F_2 = \sum_{half2} F_{core} \tag{5-9}$$

The holding force function is formulated as follows:

$$Sat(\text{hold\_for}) = \begin{cases} 1 & \text{for } F_1 \neq F_2 \\ 0 & \text{for } F_1 = F_2 \end{cases} \tag{5-10}$$

The difference between F_1 and F_2 can make the die casting to move along with one of two halves. When F_1 and F_2 are equal, an extra device would be required to remove the die casting from the cover die. In this case, the *Sat*(hold_for) is a zero value, and this would make the die more complex. This function is applied in the case without use of side cores.

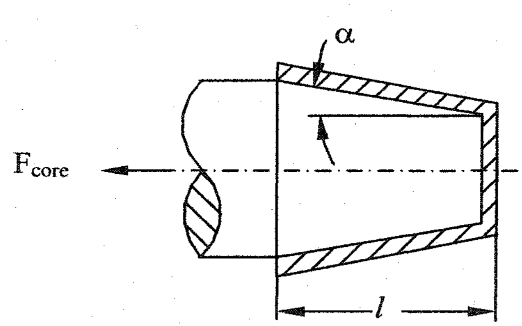


Fig. 5.3 The parameters for calculation of F_{core}

5.3.2 Test function for design constraints

For design constraints, a test will be performed so as to determine whether a constraint is satisfied or violated. The value of the test is assigned as 0 or 1 and recorded by function

Sat(const). A value of 1 indicates violation, and a value of 0 indicates satisfaction for a given constraint. The test function of design constraints is shown as follows:

Function 1 *Test function for datum plane*

If the parting plane does not include or traverse a datum plane, the rule is satisfied. The test function for datum plane is given as follows:

$$\mathrm{Sat}(dp) = \begin{cases} 0 & \text{if } dp \cap \mathrm{PS} = \{\phi\} \\ 1 & \text{otherwise} \end{cases} \tag{5-11}$$

where *dp* denotes a datum plane on a die casting and PS is the parting plane.

Function 2 *Test function for machining surface*

For the machining requirement, the following test will be performed:

$$\mathrm{Sat}(mp) = \begin{cases} 0 & \text{if } mp \subset \mathrm{PS} \\ 1 & \text{otherwise} \end{cases} \tag{5-12}$$

where *mp* denotes a plane with machining requirement, $mp \subset \mathrm{PS}$ denotes that *mp* is included or coincides with a parting plane in the design.

Fuction 3 *Test function for accuracy surface*

This can be tested by checking if the part is in the same die half:

$$Sat(ha) = \begin{cases} 0 & \text{if } ha \cap PS = \{\phi\} \\ 1 & \text{otherwise} \end{cases} \tag{5-13}$$

where *ha* denotes the part of the casting with high requirement on dimensional accuracy, and $ha \cap$PS=$\{\phi\}$ indicates part of casting does not intersect with parting plane, i.e., the part requiring high dimensional accuracy is placed in the same die half.

Function 4   *Test function for parting mark*

Similarly, the following test is required to determine if there is a parting line mark on a casting surface or not:

$$Sat(pm) = \begin{cases} 0 & \text{if } pm \cap PS = \{\phi\} \\ 1 & \text{otherwise} \end{cases} \tag{5-14}$$

where *pm* denotes a plane on which a parting line mark is unacceptable, and $pm \cap$PS=$\{\phi\}$ indicates that *pm* does not intersect with the parting plane.

5.4 Evaluation and optimization of parting schemes

In the last subsection, evaluation on a parting scheme is formulated as a number of satisfaction and test functions. Building such functions is an essential task in many design applications, particularly in situations where the design solution is obtained through optimization techniques [Balachandran 1993]. In this section, the evaluation and optimization of parting schemes are discussed.

Each satisfaction function reflects an aspect of parting scheme design. If the overall performance of a parting scheme is determined by accumulating the value of each satisfaction function, then Equation (5-1) can be written as follows:

$$\xi\,(\mathrm{CPS}_j) = \sum_{i=1}^{5} Sat_j(\mathrm{critr}_i) \tag{5-15}$$

Finally, the evaluation on performance of each parting scheme can be obtained by the sum of values of each *Sat*(critr) and a number of violated constraints.

In the parting scheme design, the design rules should be satisfied as much as possible. The maximal satisfaction of design rules is the goal of the parting scheme design. In the evaluation of a parting scheme, the design criteria are modelled as satisfaction functions and the design constrains are presented as test functions. The value of each function indicates the degree of satisfaction for a given criterion or the violation of a constraint. The summation of values from each satisfaction function gives the overall evaluation for a parting scheme, i.e. the performance of a parting scheme is determined by accumulating the values of each satisfaction function. Obviously, the satisfaction function corresponds to the objective function, and the test function corresponds to the constraint function in optimization procedure. The optimization of a parting scheme is to find a parting scheme with maximum value of objective function and without any violation of the design constraints. The formulation of the optimization can be described as:

$$\text{Maximize} \qquad \xi(\text{CPS}_j) = \sum_{i=1}^{5} Sat(\text{crit}_i) \tag{5-16}$$

$$\text{Constraints} \qquad Sat(\text{const}_k)=0$$

Where j is the number of candidate parting schemes

i is the number of satisfaction functions and

k is the number of design constraint

The candidate parting scheme with maximum value of objective function is regarded as the optimal parting scheme. Since the constraints in a parting scheme design are not always mandatory, $Sat(\text{const}_k)$ function is mainly used for verifying whether a design constraint is satisfied or violated, and provides the judgment information for the selection of a scheme.

The effect of design criteria may be different for different designs. If a certain rule is more important, the rule should be stressed. Otherwise, the rule can be relaxed. In order to balance or synthesize the effect of each satisfaction function, a weighted optimization is introduced. The weighted optimization formula can be written as:

$$\text{Maximize} \qquad \xi(\text{CPS}_j) = \sum_{i=1}^{5} c_i Sat(\text{crit}_i) \tag{5-17}$$

where c_i is the weighted coefficient for criterion i. The weighted coefficients are assigned between 0 and 1. This allows the designers to emphasize or de-emphasize the importance of certain criteria.

Linear programming technique cannot be readily applied to maximize the objective function $\xi(\text{CPS}_j)$ as the constraints cannot be described with a set of linear inequalities.

5.5 Case analysis

In order to demonstrate the evaluation and optimization of parting schemes, the parting schemes generated for a support frame die casting as shown in Fig. 4.8, Fig.4.13, Fig. 4.15 and Fig. 4.16 in Chapter 4 are used as a case. To demonstrate the effect of geometric size on the parting scheme selection, two sets of sizes are used to define the casting. The first set of sizes of the support frame casting is given in Table 5.3, and the volume of each CRs is given in Table 5.4. Symbols used in Table 5.3 are defined in Fig. 4.8 of Chapter 4. There are three candidate parting schemes, called scheme A, scheme B and scheme C, corresponding to the scheme shown in Fig. 4.13, Fig. 4.15 and Fig. 4.16 respectively.

Table 5.3 The first set of sizes of support frame die casting (cm)

a	b	c	d	e	f	g	L	w	h	R	α
5	0.3	2	0.3	2	42	0.3	50	10	30	100	24°

Table 5.4 Volume of eight CRs (cm^3)

CR_1	CR_2	CR_3	CR_4	CR_5	CR_6	CR_7	CR_8
596.00	482.16	482.16	5602.15	206.46	5602.15	275.50	275.50

Applying the five satisfaction functions, the performances of the candidate parting schemes are listed in Table 5.5 and Table 5.6.

Table 5. 5 *Sat*(num_sc) and *Sat*(vol_sc) of parting schemes

Scheme	N	n	$\sum_{j=1}^{N} v_j$	$\sum_{i=1}^{n} v_i$	Sat(num_sc)	Sat(vol_sc)
Scheme A	8	8	13522	13522	1.000	1.000
Scheme B	8	4	13522	12006	0.500	0.888
Scheme C	8	4	13522	1516	0.500	0.112

Table 5.6 *Sat*(ps), *Sat*(cast_h) and *Sat*(hold_for) of parting schemes

Scheme	A_{ps}	A_{pro_ps}	h_{max}	H_{max}	F_1, F_2	Sat(ps)	Sat(cast_h)	*Sat*(hold_for)
Scheme A	1425	1425	5	40	=	1.000	0.889	0
Scheme B	500	500	30	30	≠	1.000	0.500	1
Scheme C	521	300	28.5	20	≠	0.576	0.412	1

Suppose that there is a constraint on the top surface of primitives ① and ② and ③, i.e., the parting line marker is unacceptable on them. It means that the parting plane should not traverse the top surfaces. Then four test functions are activated and the results are given in Table 5.7.

Table 5.7 The results of four test functions

Scheme	Sat(dp)	Sat(mp)	Sat(ha)	Sat(pm)
Scheme A	-	-	-	1
Scheme B	-	-	-	0
Scheme C	-	-	-	0

Overall performance of each scheme is given below

(Scheme A): $\xi(CPS_A)=3.889$;

(Scheme B): $\xi(CPS_B)=3.888$;

(Scheme C): $\xi(CPS_C)=2.600$;

Obviously, the function value for Scheme A is the highest, Scheme C is the lowest and Scheme B is in the middle. The weighted optimization is used to determine the final scheme.

Among the satisfaction functions, *Sat*(num_sc) and *Sat*(vol_sc) are used to measure the satisfaction degree of the side cores and both are separately counted. This results in an overestimation of the effect on the use of side cores. In general, *Sat*(vol_sc) is often used as the major criterion for minimal use of side cores. *Sat*(num_sc) can therefore be weakened. Besides, *Sat*(hold_for) can only take zero or one, the effect of the holding force will have an excessive influence on the evaluation result. The effect of the holding force can be reduced by adjusting the weighting factors. In fact, the selection of the weighting factors is based on the experience of die designer. Each factor can be selected between 0 and 1. The weighting factors used in this optimization are given in Table 5.8.

Table 5.8 The weighting factors used in the optimization

c_{num_sc}	c_{vol_sc}	c_{ps}	c_{cast_h}	c_{hold_for}
0.30	0.80	0.90	1.00	0.20

The results of optimization are as follows:

(Scheme A): $\xi(CPS_A)=2.889$;

(Scheme B): $\xi(CPS_B)=2.460$;

(Scheme C): $\xi(CPS_C)=1.365$;

The weighting changes the overall performance value although the order of the performance value remains unchanged in the given casting. The optimal scheme is chosen to be scheme A, as shown in Fig. 5. 4.

The optimal parting scheme is also influenced by the geometrical size although the

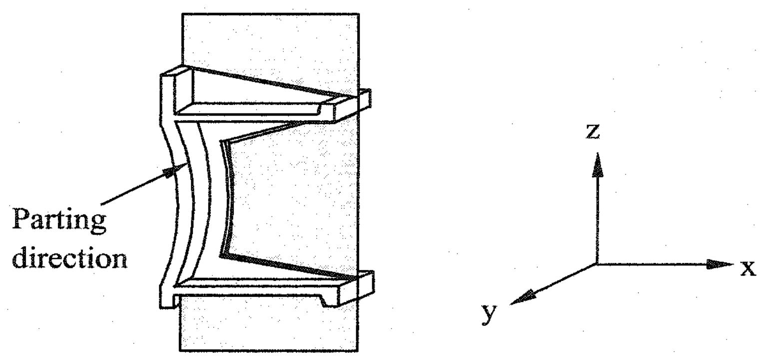


Fig 5.4 Scheme A chosen as optimal parting scheme

primitives and topological relationship are identical. To illustrate the effect of geometric size on the optimal parting scheme, the second set of size of the same casting (i.e., the support frame) is varied and the dimension is defined in Table 5.9. The volumes of the eight IEs are given in Table 5.10.

Table 5.9 The second set of sizes of support frame die casting (cm)

A	B	C	d	E	F	G	L	w	H	R	α
5	0.3	2	0.3	2	42	0.3	50	50	10	100	24°

Table 5.10 Volume of eight CRs (cm^3)

CR_1	CR_2	CR_3	CR_4	CR_5	CR_6	CR_7	CR_8
2980.00	779.63	779.63	8400.15	21.46	8400.15	445.50	445.50

The performances of the candidate parting schemes are listed in Table 5.11 and Table 5.12.

Table 5.11 *Sat*(num_sc) and *Sat*(vol_sc) of parting schemes

Scheme	N	n	$\sum_{j=1}^{N} v_j$	$\sum_{i=1}^{n} v_i$	Sat(num_sc)	Sat(vol_sc)
Scheme A	8	8	22525	22525	1.000	1.000
Scheme B	8	4	22525	19801	0.500	0.889
Scheme C	8	4	22525	2724	0.500	0.121

Table 5.12 *Sat*(ps), *Sat*(cast_h) and *Sat*(hold_for)of parting schemes

Scheme	A_{ps}	A_{pro_ps}	h_{max}	H_{max}	F_1,F_2	Sat(ps)	Sat(cast_h)	*Sat*(hold_for)
Scheme A	490	490	25	30	=	1.000	0.545	0
Scheme B	2500	2500	10	50	$\neq$	1.000	0.833	1
Scheme C	1575	500	28.5	30	$\neq$	0.317	0.513	1

The result of the four test functions is the same as those listed in Table 5.7. The total performance of each parting scheme for given size is as follows:

(Scheme A): $\xi(CPS_A)=3.545$;

(Scheme B): $\xi(CPS_B)=4.222$;

(Scheme C): $\xi(CPS_C)=2.451$;

The result of the optimization is as follows:

(Scheme A): $\xi(CPS_A)=2.545$; ,

(Scheme B): $\xi(CPS_B)=2.794$;

(Scheme C): $\xi(CPS_C)=1.245$;

The optimal scheme for the second set of data is found to be Scheme B, as shown in Fig. 5. 5.

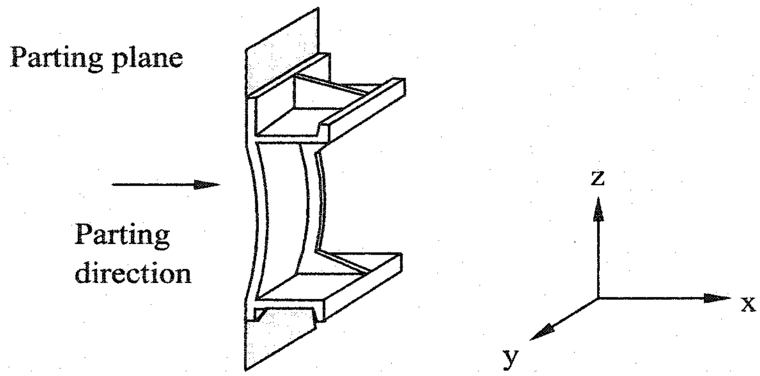


Fig. 5.5 Scheme B chosen as optimal parting scheme

Obviously, different optimal schemes are generated for different geometrical size. There is a good agreement between the schemes chosen and the real design practice based on experience.

The dimensions of the second example, a casing body as shown in Fig. 4.17, are given in Fig. 5.6. The evaluation and optimization results are given in Table 5. 13. According to the result, the first parting scheme shown in Fig.4. 19 (a) is selected as the parting scheme.

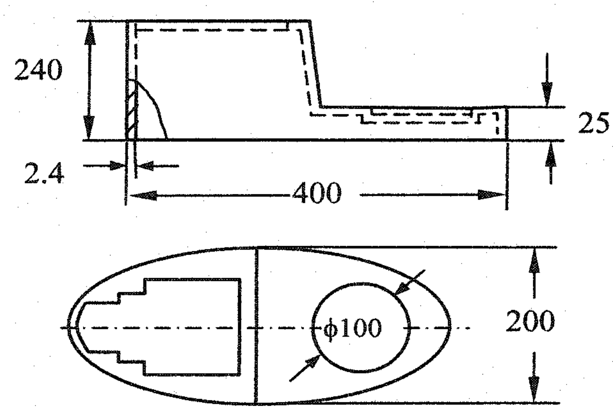


Fig. 5.6 Dimension of the casing body

Table 5. 13 Evaluation and optimization results of the casing body casting

Scheme	Scheme (a)	Scheme (b)
Sat(num_sc)	1.00	1.00
Sat(vol_sc)	1.00	0.40
Sat(ps)	1.00	1.00
Sat(cast_h)	0.67	0.73
Sat(hold_for)	1.00	1.00
ξ(CPS)	4.67	4.13
Optimization	2.87	2.45

5.6 Evaluation and optimization process control

After all the parting schemes have been generated in the stage of scheme generation, the control module will activate the scheme evaluation routine to evaluate the performance of each candidate parting scheme. After all aspects of the performance of each parting scheme have been obtained in the stage of the scheme evaluation, the optimization routine is invoked to optimize the candidate parting schemes. The control flow chart is illustrated in Fig. 5.7

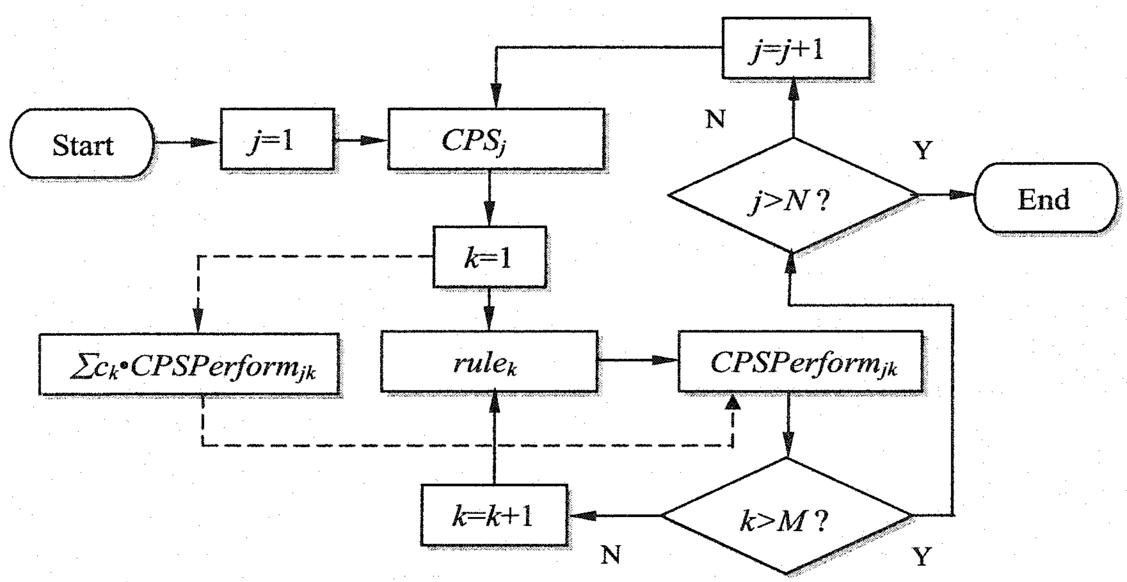


Fig. 5. 7 The control flow chart of parting scheme evaluation

Where CPS_j is jth candidate parting scheme, $rule_k$ is the kth evaluation model and CPSPerform$_{jk}$ is evaluation result for CPS_j on $rule_k$. N denotes the total number of CPS, and M denotes the number of evaluation models. After a CPS is sent to the evaluation routine, the evaluation action can be activated. The evaluation model is called one by one until they are exhausted. Each performance is stored. After the evaluation operation is completed in one CPS, the next one will be sent and the next evaluation will be performed. This procedure is repeated until all CPS are exhausted. In this way, the performance of each candidate parting schemes is generated. The dashed line denotes the weighted optimization procedure. The $\sum c_k \bullet CPSPerform_{jk}$ denotes the accumulation of the values of the weighted performance of the jth candidate parting scheme. The accumulation operation is done one by one for the parting scheme. At last, the optimal results are listed in the order of magnitude of scheme performance, and the final scheme can be determined.

5.7 Summary

This chapter describes a system for the parting scheme evaluation and optimization. The major components of the system are as follows:

i) The rules commonly used in parting scheme design are described. These rules are used for the evaluation of candidate parting schemes.

ii) Based on the design rules, an evaluation model is developed. The five design criteria are formularized as satisfaction function and four design constraints are formularized as the test functions. From the value of the functions, the scheme is evaluated.

iii) Rule-based evaluation and optimization are proposed and three examples are given to demonstrate the process of evaluation and optimization.

CHAPTER 6

PARAMETRIC DESIGN OF A DIE CASTING SCHEME

6.1 Basic description

Selection of die casting parameters is an important phase of die casting scheme design. There are many process parameters in die casting, such as filling velocity, filling time, injection pressure and so on. These parameters influence gating design and die casting quality. Correct selection of the parameters is a challenging work in die casting scheme design and die casting production. In order to achieve the correct selection of the parameters, much research has been done on fluid dynamics and heat transfer problems [Louis and Draper 1966, Kopf 1968, Lindsey and Wallace 1968, Thukkaram 1972, Herman 1973, Akira and Tatsuichi 1985, Davey and Hinduja 1991]. Many experimental data, diagram and formulas for the selection of design parameters stem from this research work. Generally, the data are presented in the form of charts, diagrams, formulae and so on. Table 6.1, Table 6.2 and Fig. 6.1 give the typical format of the experimental data and diagrams for the selection of filling parameters in die casting handbook [Zhan 1989, Shen 1982, Shi and Wang 1988, Li 1997, Venus 1975]. According to a required characteristic of die casting, a particular value of parameter is selected from the list, diagram or use of simple formula. There are three difficulties in selecting die casting parameters in a such way. These are:

i) Difficulty to address a specific value due to a large range of data,

ii) Difficulty to work consistently due to the difference of the experience of individual designer, and

iii) Difficulty to consider multiple factors due to separated data.

In order to overcome the above drawbacks, a fuzzy synthetic evaluation technique is introduced for the determination of the die casting parameters in this chapter. The fuzzy algorithms outperform conventional methods when they are appropriately applied to solve the problems of fuzzy uncertainty [Zheru et al 1996]. Many researches show that the application of fuzzy synthetic evaluation in the selection of design parameters can provide more reasonable and economical results since the fuzzy synthetic evaluation algorithm can quantitatively and synthetically handle various fuzzy factors [Wang 1992]. In fact, the selection of die casting parameters presents fuzzy and subjective characteristic, and is often based on the trial and error procedure. In selecting die casting process parameters, many factors are of fuzzy quantity, such as large or small casting size, thin or thick casting thickness, complex or simple casting shapes. A parameter can be reasonably determined only if the fuzzy quantities are reasonably estimated. However, the factors interact with one another. It is often difficult to make a reasonable and consistent choice from a given range of values assigned to the parameter (i.e., the value domain). Fuzzy synthetic evaluation divides factors and value domains of the parameter into discrete quantities to create a fuzzy factor set and alternative set of the parameter. The contribution of each discrete level and each factor can then be synthetically considered. It can be expected that fuzzy synthetic evaluation can give reasonable and consistent results provided the factor set and alternative set are completely provided [Wang and Song 1987]. This chapter will make an attempt to apply fuzzy synthetic evaluation algorithm and to provide a new method to select die casting parameters. The details are described in the following sections.

Table 6.1 Selection of filling time (s)

Wall thickness (mm)	1.5	1.8	2.0	2.3	2.5	3.0	3.8	5.0	6.4
Filling time	0.01 -0.03	0.02 -0.04	0.02 -0.06	0.03 -0.07	0.04 -0.09	0.05 -0.10	0.05 -0.12	0.06 -0.20	0.08 -0.30

Table 6.2 Selection of filling velocity (m/s)

Alloy	Casting characteristic		
	simple	average	complex
Al	10-15	15-45	30-60
Zn	10-20	15-30	20-40

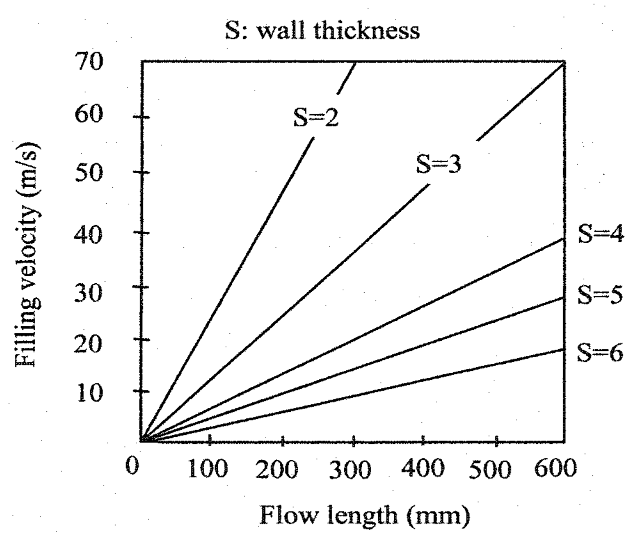


Fig. 6.1 Filling velocity at different wall thickness and flow length

6.2 Mechanism of fuzzy synthetic evaluation

Synthetic evaluation refers to an overall evaluation on an object or behaviour to be assessed which is under the influence of multiple factors. If these factors present fuzzy characteristics, the synthetic evaluation is referred to as a fuzzy synthetic evaluation. In the fuzzy synthetic evaluation, the following components are required:

i) *Factor set*

Factor set is a general set comprised of all factors influencing the object to be evaluated. Construction of a factor set requires a complete collection of the basic factors. These factors are allowed to be fuzzy.

ii) *Weighting set*

In general, the importance of each factor is different. The important factor should be emphasized and the unimportant ones should be relaxed. Therefore, a weighting set is required to represent the importance of each factor for the evaluated object. The weighting value of each factor can be determined by individual subjective judgement or by grading the membership of each factor. It should be noted that the weighting values also have an influence on the evaluation result. The weighting coefficients are non-negative and should be unitized.

iii) *Alternative set (evaluation set)*

An alternative set is a set comprised of all possible results for the evaluated object. The purpose of the fuzzy synthetic evaluation is to compose an optimal evaluation result from the alternative set.

The fuzzy synthetic evaluation is divided into two basic stages:

i) Stage1 *Single factor evaluation*

In this stage, all the factors influencing the selection of casting parameters should be defined. Each factor will be independently evaluated so as to determine the grade of membership of an evaluated object in elements of alternative set. The result of single factor evaluation is a number of evaluation sets of single factors depending on the number of factors under consideration. Elements in the evaluation set compose an evaluation matrix, which will be used in the synthetic evaluation stage.

ii) Stage 2 *Fuzzy synthetic evaluation*

Single fuzzy evaluation reflects the effect of each single factor on the object to be evaluated. Synthetic evaluation synthesizes the effect of all factors on the evaluated object. The result of synthetic evaluation is a synthetic evaluation set. The elements in the set are called a fuzzy synthetic evaluation index. The index shows the grade of membership of evaluated object in each element in the alternative set. Based on the index, a specific result can be determined by using the method of maximum grade of membership, or weighted average, or fuzzy distribution. In this stage, all factors involved are synthetically evaluated.

The mechanism above described is for one level fuzzy synthetic evaluation. For the more complex system, multiple level evaluation should be introduced. The factors should be subdivided into levels, each level also needs to be evaluated. In this work, a two level synthetic evaluation is taken. The application of fuzzy synthetic evaluation can change the conventional manner of design of die casting parameters. By fuzzy synthetic evaluation, a consistent resolution can be obtained. Fig. 6.2 shows the transformation from factors to a specific value of parameter under consideration with fuzzy synthetic evaluation. After a number of factors are input, a specific parameter value is output. Since the approach can synthesize multiple factors and multiple levels of factors, a reasonable and consistent result can be expected.

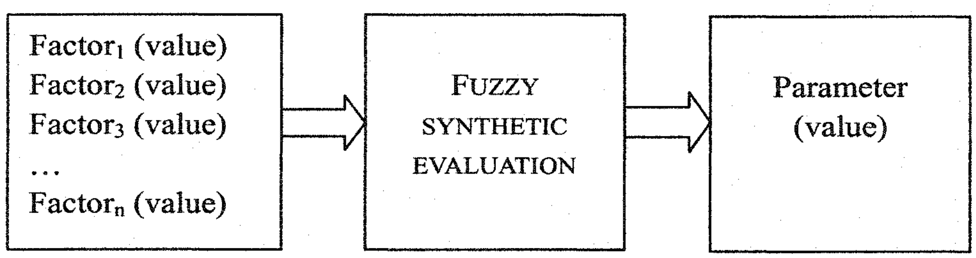


Fig. 6.2 Die casting parameter design with fuzzy synthetic evaluation

6.3 Establishment of fuzzy sets

In the selection of process parameters, a common practice is to make a choice from the data list given in a die casting handbook according to the characteristic of a die casting. In general, the characteristic of a die casting include a number of factors depending on the parameter type required. Suppose a particular parameter is required, and there are n basic factors to influence the determination of the parameter. Let u_1 denote the first factor, u_2 denote the second one and u_n denote the nth one. The factor set influencing the parameter is then defined as:

$$U=\{u_1, u_2, \cdots, u_n\} \tag{6-1}$$

The magnitude of the parameter is related with the scale of each factor. By mapping each factor influencing the parameter on a value domain $[v_b, v_t]$, each factor can be divided into levels to correspond a specific value of the parameter in domain $[v_b, v_t]$. The level subset u_i is obtained:

$$u_i=\{u_{i1}, u_{i2}, \cdots u_{ij}, \cdots, u_{im}\} \tag{6-2}$$

where u_{ij} (i=1, 2, ⋯, n; j=1, 2, ⋯, m) is jth level of ith factor. Due to the fuzziness of the factors and levels, each factor can be considered as a fuzzy subset in the universe of the level, that is

$$u_i = \frac{\mu_{i1}}{u_{i1}} + \frac{\mu_{i2}}{u_{i2}} + \cdots + \frac{\mu_{ij}}{u_{ij}} + \cdots + \frac{\mu_{im}}{u_{im}} \tag{6-3}$$

where $0 \leqslant \mu_{ij} \leqslant 1$ (i=1, 2, ⋯, n; j=1, 2, ⋯, m) is the membership grade of the jth level of the ith factor in the factor. The grade of membership of each level in a factor can be determined by using membership function [Zadeh 1978, Lee and Zhu 1995].

Suppose the filling velocity is required in a die casting scheme. The major factors which influence filling velocity include the die casting complexity, geometrical size and wall thickness. Besides, alloy, quality requirement, metal temperature and die temperature also influence the filling velocity [Venus 1975]. For simplicity, only three major factors, i.e., complexity of casting, geometrical size and wall thickness are selected to demonstrate the use of fuzzy synthetic evaluation. It should be noted that the wall thickness may be the minimal wall thickness or average wall thickness. Geometrical size may be a maximal linear size or special diagonal line of a casting. Let u_1 denote the wall thickness, u_2 denote the geometric size and u_3 denote the complexity of the casting, then the factor set is obtained:

$$U=\{u_1, u_2, u_3\} \tag{6-4}$$

Each factor is divided into five levels, the level subset u_i is obtained:

$$u_i=\{u_{i1}, u_{i2}, u_{i3}, u_{i4}, u_{i5}\} \tag{6-5}$$

Factors and levels are shown in Table 6.3.

Table 6.3 Factors and levels

Factors		Levels				
		1	2	3	4	5
u_1	wall thickness	very thin	thin	Intermediate	thick	very thick
u_2	geometrical size	very large	large	Intermediate	small	very small
u_3	shape complexity	very complex	complex	Intermediate	simple	very simple

Suppose the average wall thickness used in die castings is in a range from 0.5mm to 8mm for aluminum alloys, and 0.8mm, 2.4mm, 4.0mm, 5.6mm and 7.2mm are selected as critical values of the wall thickness corresponding to five levels, then the grade of membership of any wall thickness in the wall thickness can be calculated with max-min membership function:

$$\mu_{11} = \begin{cases} 1 & \text{for } 0.0 < u_1 \leq 0.8 \\ [(9-u_1)/8.2]^3 & \text{for } 0.8 < u_1 < 9.0 \\ 0 & \text{for } u_1 \geq 9.0 \end{cases}$$

$$\mu_{12} = \begin{cases} (u_1/2.4)^3 & \text{for } 0.0 < u_1 \leq 2.4 \\ [(9-u_1)/6.6]^3 & \text{for } 2.4 < u_1 < 9.0 \\ 0 & \text{for } u_1 \geq 9.0 \end{cases}$$

$$\mu_{13} = \begin{cases} (u_1/4.0)^3 & \text{for } 0.0 < u_1 \leq 4.0 \\ [(9-u_1)/5.0]^3 & \text{for } 4.0 < u_1 < 9.0 \\ 0 & \text{for } u_1 \geq 9.0 \end{cases} \qquad (6\text{-}6)$$

$$\mu_{14} = \begin{cases} (u_1/5.6)^3 & \text{for } 0.0 < u_1 \leq 5.6 \\ [(9-u_1)/2.4]^3 & \text{for } 5.6 < u_1 \leq 9.0 \\ 0 & \text{for } u_1 \geq 9.0 \end{cases}$$

$$\mu_{15} = \begin{cases} (u_1/7.2)^3 & \text{for } 0.0 < u_1 < 7.2 \\ 1 & \text{for } u_1 \geq 7.2 \end{cases}$$

Taking each critical factor of the above functions as an example, the grade of membership of each critical value in the wall thickness can be calculated and is given in Table 6.4.

From Table 6.4, the grade of membership of each critical wall thickness in wall thickness factor can be found. Similarly, suppose the die casting size is below 460mm, and 100mm, 190mm, 280mm, 370mm and 460mm are selected as critical values of the casting size corresponding to five size levels. Then the grade of membership of each level in the die casting size can be calculated in a similar way. The grade of membership for each critical size value is given in Table 6.5.

Table 6.4 Grade of membership for critical values of wall thickness

u_1 (mm)	μ_{11}	μ_{12}	μ_{13}	μ_{14}	μ_{15}
0.8	1.000	0.037	0.008	0.003	0.001
2.4	0.521	1.000	0.216	0.079	0.037
4.0	0.226	0.435	1.000	0.364	0.171
5.6	0.071	0.137	0.314	1.000	0.471
7.2	0.011	0.020	0.047	0.148	1.000

Table 6.5 Grade of membership for critical values of casting size

u_2 (mm)	μ_{21}	μ_{22}	μ_{23}	μ_{24}	μ_{25}
100	1.000	0.145	0.046	0.020	0.010
190	0.465	1.000	0.312	0.135	0.070
280	0.166	0.357	1.000	0.433	0.226
370	0.034	0.740	0.206	1.000	0.520
460	0.001	0.002	0.006	0.029	1.000

The complexity of a die casting is indicated by the number of entities composing the casting, and 2, 3, 4, 5, and 6 are selected as critical number of entities, then the grade of membership of each critical number in the casting complexity for each critical number is given in Table 6.6.

The fuzzy subsets of three factors in the level universe have been constructed by using the max-min function. It should be noted that there is no limitation on the number of factors, levels, magnitude of boundary and the critical values provided they are arranged in accordance with die casting practice. In fact, they can be divided more finely. So far, the fuzzy set has been established. The alternative set is discussed in the following section.

Table 6.6 Grade of membership for critical numbers of casting entities

u_3 (number)	μ_{31}	μ_{32}	μ_{33}	μ_{34}	μ_{35}
2	1.000	0.296	0.125	0.064	0.037
3	0.512	1.000	0.422	0.216	0.125
4	0.216	0.422	1.000	0.512	0.296
5	0.064	0.125	0.296	1.000	0.579
6	0.008	0.016	0.037	0.125	1.000

6.4 Establishment of alternative set

In the selection of die casting parameter, the determination of a parameter value is dependent on the scale of factors influencing the parameters. In all cases, possible values are always distributed within a specific range. In fact, the selection of parameter is to select a particular value in that range according to certain principles or rules. For example, in selecting the filling velocity, the following rules are used:

" If wall thickness is thin, then high filing velocity should be chosen;

If die casting is complex, then high filling velocity should be chosen;

If casting size is large, then high filling velocity should be chosen;

If metal temperature is high, then low filling velocity should be chosen;

… ″

These rules are general and fuzzy, and an exact solution cannot be addressed. In engineering problems, a design parameter is usually defined in a specific domain $[v_b, v_t]$ [Wang 1987]. Discretization of the domain can yield a set of discrete values, v_k (k=1, 2, ⋯, p), such a set is called an alternative set of the parameter

$$V=\{v_1, v_2, \cdots, v_p\} \tag{6-7}$$

The alternative set presents a number of elements covering all the possible evaluation results for the object being evaluated. The task of fuzzy synthetic evaluation is to select optimal results from alternative set according to the all the factors under consideration.

Suppose the filling velocity can be selected in the range between 10 m/s and 120m/s for a specific die casting machine, the value domain [10, 120] is then the range of filling velocity. The alternative set of the filling velocity can be obtained by dividing the domain into discrete values. If an interval of 10 m/s is taken, the alternative set of the filling velocity is as follows:

$$v_f=\{120, 110, 100, 90, 80, 70, 60, 50, 40, 30, 20, 10\} \tag{6-8}$$

For any parameter to be evaluated, an alternative set can be constructed.

6.5 Primary synthetic fuzzy evaluation

After the fuzzy set and the alternative set are established, the primary fuzzy evaluation can be performed. The primary fuzzy evaluation handles the fuzziness of the factors by synthesizing the contribution of each level of a factor to a parametric value. Suppose the parameter is evaluated against the jth level of ith factor, the grade of membership of the level in the kth element in the alternative set is denoted as r_{ijk} (i=1, 2, ⋯, n; j=1, 2, ⋯, m; k=1, 2, ⋯, p), then the evaluation set of each level of the ith factor can be expressed as

$$R_{ij} = \frac{r_{ij1}}{(u_{ij}, v_1)} + \frac{r_{ij2}}{(u_{ij}, v_2)} + \cdots + \frac{r_{ijp}}{(u_{ij}, v_p)} \tag{6-9}$$

Obviously, the level evaluation set is a fuzzy subset on the alternative set. The matrix comprised of the grade of membership of the evaluation set of each level of ith factor is called the level evaluation matrix of the ith factor. Mapping the factor levels in the domain of the alternative set, and using symmetrical distribution about the principal element for the grade of membership of an evaluation set in each level [Wang and Song 1987], the level evaluation matrix can then be obtained as

$$R_i = \begin{pmatrix} r_{i11} & r_{i12} & \cdots & r_{i1p} \\ r_{i21} & r_{i22} & \cdots & r_{i2p} \\ \cdots & \cdots & \cdots & \cdots \\ r_{im1} & r_{im2} & \cdots & r_{imp} \end{pmatrix} \tag{6-10}$$

According to the definition, the level evaluation matrix for a factor can be written as

$$R_i = \begin{pmatrix} 0.9 & 1.0 & 0.9 & 0.7 & 0.5 & 0.3 & 0.1 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.3 & 0.5 & 0.7 & 0.9 & 1.0 & 0.9 & 0.7 & 0.5 & 0.3 & 0.1 & 0.0 & 0.0 \\ 0.0 & 0.1 & 0.3 & 0.5 & 0.7 & 0.9 & 1.0 & 0.9 & 0.7 & 0.5 & 0.3 & 0.1 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.1 & 0.3 & 0.5 & 0.7 & 0.9 & 1.0 & 0.9 & 0.7 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.1 & 0.3 & 0.5 & 0.7 & 0.9 & 1.0 \end{pmatrix} \tag{6-11}$$

In order to balance the effect of each level on the selection of parameter, a proper weighting coefficient can be introduced into each level. The unitary value of grade of membership for *j*th level of *i*th factor, μ_{ij} (i=1, 2, $\cdots n$; j=1, 2, $\cdots$, m), is introduced as the weighted value of the level, i.e.,

$$a_{ij} = \mu_{ij} / \sum_{j=1}^{m} \mu_{ij} \tag{6-12}$$

Then the level weight set of the *i*th factor can be obtained as

$$A_i = (a_{i1}, a_{i2}, \cdots, a_{im}) \tag{6-13}$$

The primary fuzzy evaluation set can be obtained as follows

$$B_i = A_i \bigcirc R_i = (b_{i1}, b_{i2}, \cdots, b_{ip}) \tag{6-14}$$

The symbol $\bigcirc$ denotes the fuzzy operator. The operator introduces ($\cdot$, +) algorithm, thus

$$b_{ik} = \sum_{j=1}^{m} a_{ij} \cdot r_{ijk} \tag{6-15}$$

In regard to the contribution of each level of the *i*th factor, b_{ik} is the grade of membership of parameter object in the *k*th element in alternative set. The matrix comprised with elements b_{ik} is denoted as the level synthetic evaluation matrix or primary synthetic evaluation matrix. Suppose the casing body casting taken as example in Chapter 4 has the following characteristics: wall thickness: 2.4mm, casting size: 460mm (spacial diagonal line), entity number: 2. The grade of membership of each level for the three factors is shown in Table 6.7. The weight sets are as follows:

$$A_1=(0.281, 0.540, 0.117, 0.043, 0.020), \tag{6-16}$$

$$A_2=(0.001, 0.002, 0.006, 0.028, 0.963), \tag{6-17}$$

$$A_3=(0.669, 0.180, 0.084, 0.043, 0.025). \tag{6-18}$$

The primary synthetic evaluation matrix for the given case is

$$R=\begin{pmatrix} B_1 \\ B_2 \\ B_3 \end{pmatrix}=\begin{pmatrix} 0.4140 & 0.5627 & 0.6600 & 0.7412 & 0.7667 & 0.6005 & 0.4413 & 0.4114 & 0.2926 & 0.1695 & 0.0918 & 0.0610 \\ 0.0015 & 0.0026 & 0.0041 & 0.0055 & 0.0095 & 0.0159 & 0.1178 & 0.3149 & 0.5115 & 0.7053 & 0.8937 & 0.9832 \\ 0.6561 & 0. 7674 & 0.7533 & 0.6723 & 0.5776 & 0.4512 & 0.3009 & 0.2032 & 0.1640 & 0.1043 & 0.0864 & 0.0635 \end{pmatrix} \tag{6-19}$$

Table 6.7 Grade of membership of each level for three factors

Factors	Grade of membership				
	1	2	3	4	5
u_1 (2.4mm)	0.521	1.000	0.216	0.079	0.037
u_2 (460mm)	0.001	0.002	0.006	0.029	1.000
u_3 (2)	1.000	0.269	0.125	0.064	0.037

Once the primary synthetic evaluation matrix has been obtained, the secondary synthetic evaluation can be performed.

6.6 Secondary fuzzy synthetic evaluation

The primary fuzzy evaluation integrates the contributions of each level, and embodies the effect of each factor in the selection of parameter. Usually, the effect of each factor on the parameter selection is different, and the weighted value should be required. Suppose a_i (i=1, 2, 3, ⋯, n) is the weighted value of the ith factor, then the weighted set of factors can be written as

$$A=(a_1, a_2, \cdots, a_n) \tag{6-20}$$

Where a_i is positive and $\sum_{i=1}^{n} a_i = 1$. The set of secondary synthetic evaluation against all factors is obtained as

$$B=A \bigcirc R=(a_1, a_2, \cdots, a_n) \bigcirc \begin{pmatrix} b_{11}, & b_{12}, & \cdots, & b_{1p} \\ b_{21}, & b_{22}, & \cdots, & b_{2p} \\ \cdots, & \cdots, & \cdots, & \cdots \\ b_{n1}, & b_{n2}, & \cdots, & b_{np} \end{pmatrix}$$

$$=(b_1, b_2, \cdots, b_p) \tag{6-21}$$

Where $b_k = \sum_{i=1}^{n} a_i \cdot b_{ik}$, b_k is the grade of membership of parameter object in the kth element of alternative set under consideration of the effect of all factors. The synthetic evaluation set B will serve as evaluation criteria for the selection of parameter value.

Taking b_k as the weighted coefficient, the value of the parameter required can be finally determined by the weighted average on v_k (k=1, 2, ⋯, p), i.e.,

$$v_f = \sum_{k=1}^{p} b_k \cdot v_k \Big/ \sum_{k=1}^{p} b_k \tag{6-22}$$

With the casing body example, the secondary fuzzy synthetic evaluation can be carried out and the final value of filling velocity can be obtained. Suppose the weighted set of factors is

$$A=(0.4, 0.32, 0.28) \tag{6-23}$$

Then the secondary synthetic evaluation set is

$$B=(0.4, 0.32, 0.28) \circ \begin{pmatrix} 0.4140 & 0.5627 & 0.6600 & 0.7412 & 0.7667 & 0.6005 & 0.4413 & 0.4114 & 0.2926 & 0.1695 & 0.0918 & 0.0610 \\ 0.0015 & 0.0026 & 0.0041 & 0.0055 & 0.0095 & 0.0159 & 0.1178 & 0.3149 & 0.5115 & 0.7053 & 0.8937 & 0.9832 \\ 0.6561 & 0.7674 & 0.7533 & 0.6723 & 0.5776 & 0.4512 & 0.3009 & 0.2032 & 0.1640 & 0.1043 & 0.0864 & 0.0635 \end{pmatrix}$$

$$=(0.3497, 0.4408, 0.4762, 0.4865, 0.4704, 0.3706, 0.2984, 0.3219, 0.3266, 0.3227, 0.3469, 0.3568) \tag{6-24}$$

According to equation (6-21), the filling velocity is obtained as

$$v_f = \sum_{k=1}^{12} b_k \cdot v_k \bigg/ \sum_{k=1}^{12} b_k = 312/4.5 \cong 70\ (\text{m/s}) \tag{6-25}$$

For the given die casting, a filling velocity of 70 m/s is determined. Similarly, the filling time can be determined by establishing the factor set of thickness, casting size and temperature and establishing the alternative set of filling time. The filling time for the given casting is 40 ms. With fuzzy synthetic evaluation, the factors influencing the parameter object is divided into levels, and evaluation on parameter object is performed against levels.

The fuzziness of factors can be well processed and the effect of each factor can be synthetically reflected. The fuzzy synthetic evaluation does not restrict the number and characteristics of the factors, but can present reasonable result provided the factors influencing the parameter are completely defined. Since there is no limitation to the number and characteristics of factors, the elements in the alternative set are also not limited in number and characteristic. Due to the use of grade of membership and weighted value, the fuzzy synthetic evaluation involves the subjective factor of individual person. However, the subjective factor is bounded within a small range since dual evaluation is employed in levels and factors. The validity of the synthetic evaluation approach is reliable.

6.7 Application of fuzzy synthetic evaluation

The flow chart of the program is shown in Fig. 6. 3. In general, a system program can be used without any extra effort. The system setup can be changed by the user so as to add the user's own experience for special cases, such as factors and levels, upper limitation and lower limitation of the alternative set. The factor scales can be input by the user or obtained from a file of the casting part. The system can provide an exact solution. The general format is an input/output structure as shown in Table 6.8.

Table 6.8 General format of input/output

Factors	Scale (input)	Output
?$Factor_1$	_____	
?$Factor_2$	_____	Parameter value
...		
?$Factor_n$	_____	

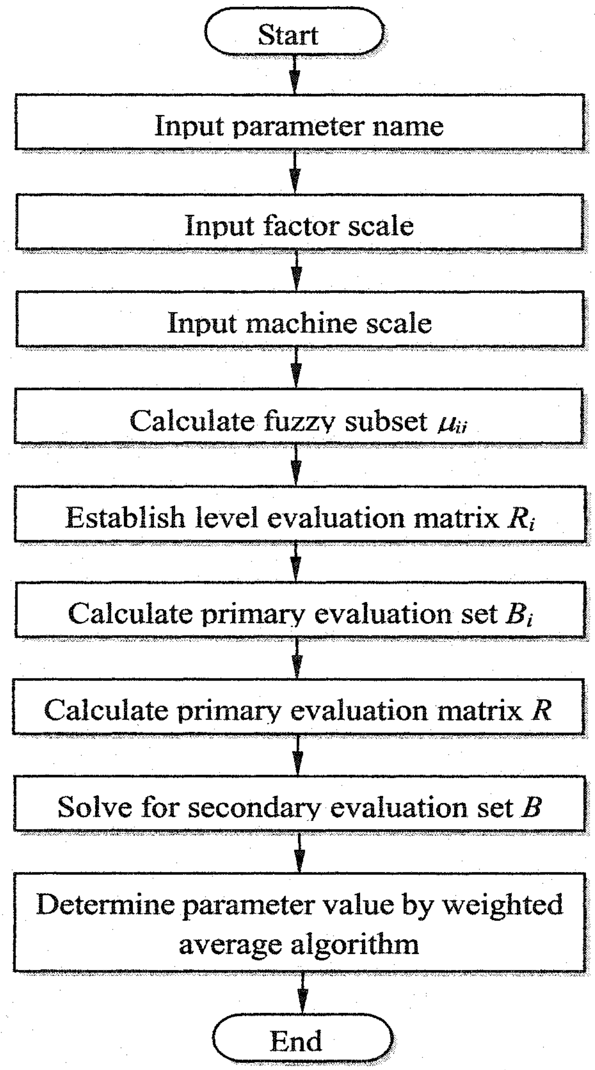


Fig. 6.3 The flow chart of the program of parameter design

For instance, the determination of filling velocity requires alloy type, casting surface finish, die and metal temperature in addition to dimensions and wall thickness. The output is a value of filling velocity. Since the system allows to change factors, levels, upper limitation and lower limitation of the alternative set, the resolution can be improved in specific situations. Compared with conventional ways, the fuzzy evaluation can provide consistent and reasonable resolution. Sometimes, there is a considerable difference between the values recommended by different die casting experts [Venus 1975]. For example, for the given casing body casting, the filling velocity recommended by Lieby is 140m/s, 45m/s by Pokorny, and 60 m/s by Detering and GroBomann. The value suggested from the present fuzzy evaluation is 70 m/s, which is lower than Lieby's, and accords with Groβmann who suggested to use half lower filling value. A high filling velocity can cause turbulent flow and air entrapment. It is desirable to use a low filling velocity whenever possible as this can avoid the formation of porosity in a casting and excessive erosion of a die cavity. It should however be noted that a too low filling velocity often results in cold defects in a die casting, such as a incomplete die filling, cold shut, and cold mark. The trial production of the given example shows that 70 m/s is reasonable and 50 m/s is a little low. In fact, traditional selection of casting parameters only involves one or two factors, the fuzzy approach can consider multi factors. From the trial result, it appears that the system can provide reasonable and consistent resolution for casting parameter design.

6.8 Summary

In this chapter, the following issues are addressed:

i) The conventional method of casting parameter design is discussed, and its drawback in selection of casting parameter is analyzed.

ii) A fuzzy synthetic evaluation approach for die casting parameters is proposed. The advantages and the principle of the fuzzy synthetic evaluation in selection of the casting parameters are described. Establishment of the fuzzy set, weighted set and the alternative set are presented.

iii) The evaluation process is provided and a case is given to show the application of fuzzy synthetic evaluation algorithm in selection of the die casting parameters.

CHAPTER 7

CONFIGURATION OF A GATING SYSTEM

7.1 Basic description

Gating provides a passage of molten metal from the shot chamber to fill up the die cavity. The gating system is of particular importance in die casting scheme design as it influences the filling pattern of molten metal and the quality of the die castings. After the parting scheme and filling parameters are determined, the gating design can be carried out. In general, the gating system includes three basic elements. They are

i) Ingate,
ii) Runner, and
iii) Sprue

Molten metal is firstly injected in the sprue, then flows through the runner and enters into the cavity through the ingate. There are three basic tasks in the gating system design phase, these are

i) Determination of the gating position
ii) Selection of the type of runners and ingates
iii) Definition of the dimensions of the gating elements

The gating position determines where the molten metal flows into the cavity. The gating type determines which form of gating system is employed, and the dimensions of gating elements determines the size scale corresponding to the casting dimensions. In the past, most work focused on the third task of gating design with purpose to reduce tedious calculation and graphical generation. The essential design features such as the determination of gating position and the selection of gating type have not been fully addressed, as there is no mathematical model to calculate them. In general, the calculation and graphical generation can be realized by conventional algorithm, and the knowledge of gating design is hard to codify and represent in an algorithm program. In this chapter, a knowledge-based technique is introduced in the development of the gating design system. The case-based approach and feature-based CAD approach are combined to generate a gating configuration. The framework of gating design is shown in Fig .7.1. There are two ways to start with the design, the left branch works in the case-based approach, and the right branch works in the feature-based CAD approach. The case-based approach is used to match the successful cases of the gating configuration by case reasoning. With reference to the matched case, a new gating system can be designed. The feature-based CAD approach is used to design a gating system under an interactive CAD environment based on the casting feature and gating knowledge. When a gating design is fired, a search in case library is performed. If the new case can match one in the library, the design proceeds in a case-based way. Otherwise, the design is performed in the feature-based CAD way.

7.2 Basic principle of feature-based gating CAD

Traditionally, software packages of gating design focused on the calculation of gate dimensions in a program under the assumption that a user knows enough about the gating knowledge and is competent to design a gating system. The use of the software is limited to

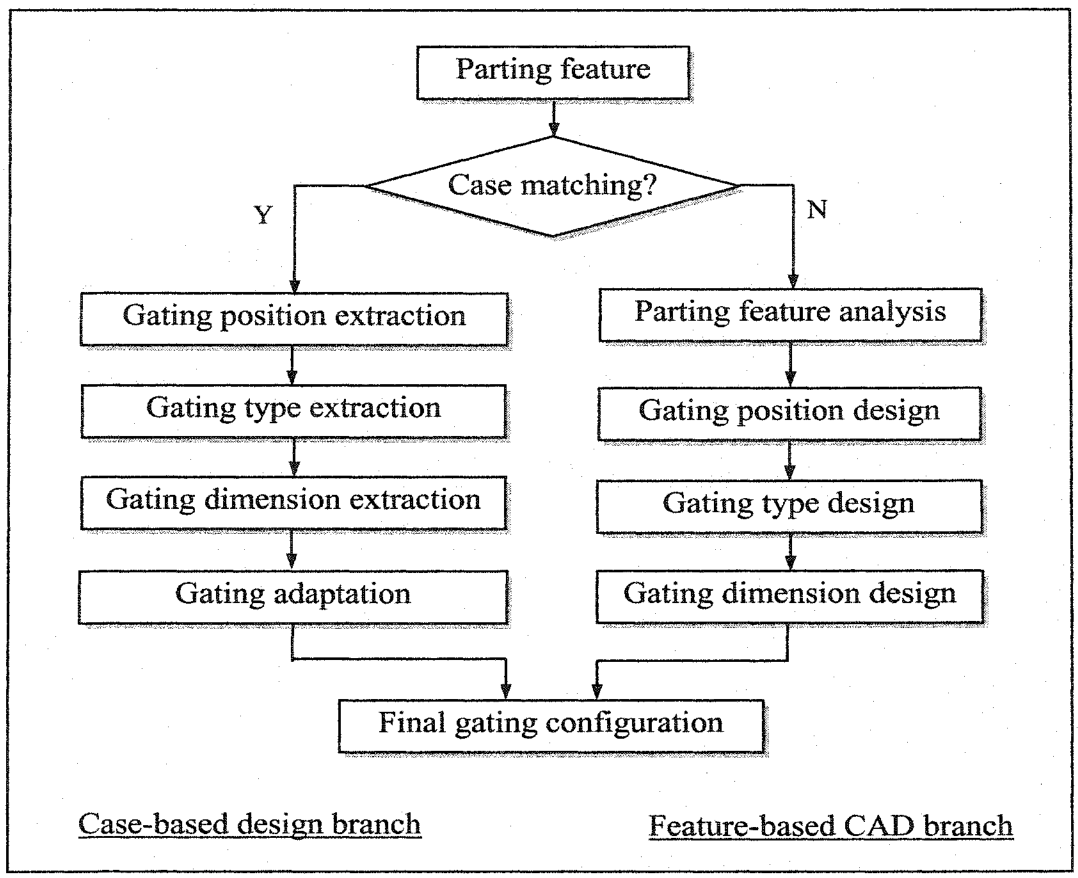


Fig .7.1 The framework of gating design

professionals who are knowledgeable in die design. In practice, users may not be aware of the best gating solution. In order to improve the shortcoming of traditional packages, a feature-based CAD approach is needed to build a support system capable of providing suitable design knowledge and suggesting the gating configuration.

7.2.1 Knowledge representation

Gating design needs to analyze information of parting feature, such as primitive type, size and necessary topological relationship, and to use design knowledge, such as gating position rules, gating type rules and so on. An appropriate knowledge representation scheme is required for achieving high performance of a knowledge-based gating system. According to the characteristics of the knowledge required in gating design, a hybrid scheme combining rule-based approach and object-oriented technique is used. The concept of knowledge object is introduced which can capture gating features and support feature management [Andrew and Wang 1994].

Production rules are a subset of predicate clauses with condition and action parts. They have significant expressive power for the range of domain-dependent inference and action specification, and can be used to represent gating rules. However, they are inadequate in linking with the gating feature and the gating configurations. According to Andrew and Wang [1994], an object is a separately defined component representing a specific item of knowledge in terms of data and procedures. The major advantage of object-orientated representation is that it provides a structural representation of useful relations, and supports a definition-by-specialization technique that is easy to use by domain experts.

Although the objects have internal data structure representing the states and methods that take actions, they do not have reasoning capability to determine which action should be taken or what decision can be made. Thus objects need knowledge to guide their behavior. Therefore, the rules are inserted into the object for this purpose. By inserting production rules into conventional object, the object contains specific state and methods and specific rules. Such an object is called a knowledge object which is a conventional object with attached rules as shown in Fig. 7.2. In the development of gating design system, production rules are used to control reasoning (knowledge of gate position and gate type) and objects are used as database that contains the data and knowledge required for reasoning (parting feature information). With the concept of knowledge object, a gating feature can be represented as a feature object that includes data, procedures and rules. The feature representation based on the knowledge object enhances the flexibility and efficiency for gating configuration.

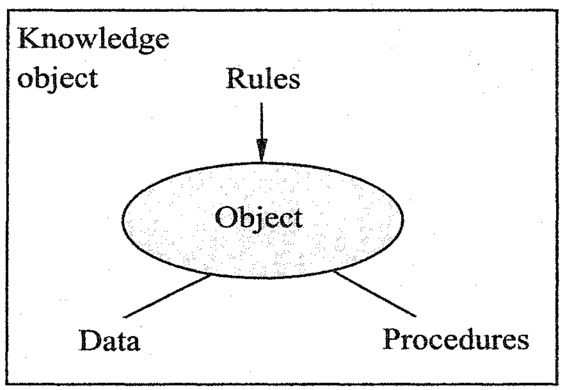


Fig. 7.2 Knowledge object

The knowledge representation in gating configuration is based on the concept of knowledge object, its general structure is as follows:

<u>Knowledge Object</u>

Data Section

data 1

...

data i

Rule Section

rule 1

...

rule j

Procedure Section

procedure 1

...

procedure k

Result Section

solution 1

...

solution m

The data in data section includes the all information to be used to infer a specific solution, such as parting primitive type, parting primitive size and parting position for inferring gating position. The data (parting feature) can be obtained from parting scheme design stage. The rule section includes rules to infer solutions based on the data given in the date section. The procedure section gives the set of program segments for performing a specific target or design action. For a given inference, there may exist several possible solutions. The knowledge object does not only include the state of the object, but also possesses the knowledge to infer the solution based on the state of the object.

As discussed earlier, there are three basic tasks in gating configuration, namely, determination of gating position, selection of gating type and definition of gating dimensions. Using the concept of a knowledge object, each feature of gating configuration can be represented with a feature object that includes data, rules, procedures and results. The required data is different for different feature objects. For example, the selection of gating position is based on the parting feature, the type, sizes and orientation of the parting primitive. Similarly, the rules and procedures that are incorporated into the knowledge object are those feature dependent. For example, the rules are gating type knowledge for the selection of gating type and the procedures are used to calculate size feature for defining gating dimension. Feature object is relatively independent and interactive by sending messages (data). As a message, the results of the current feature will be transferred to successive feature object. The data section of a feature object receives the required data, and the inference can be performed to invoke procedures or determine a solution.

7.2.2 Architecture of feature-based gating CAD

The elements of the proposed system for the gating design are shown in Fig. 7.3. The inference unit is used to solve for the local solutions. The first design phase is the determination of gating position. The determination of gating position is based on the parting feature which is stored in a separate file and is invoked into working memory at the beginning of the design. After the gating position is inferred, the inferred result is stored in the working memory and will be used as feature data in the second phase of gating type selection.

According to the type of gating being selected, gating dimensions can be defined. The gating position and gating type are the feature data required in gating dimension phase. The working memory is used to store intermediate solutions or information and provide feature data for the next stage during the solution-solving process. Each knowledge object can store and retrieve current information from the working memory. The knowledge base contains the knowledge for each stage of gating configuration. In building a gating configuration, there are some phases the system cannot make a decision independently. In a such case, an interactive mode is required. The rule base stores the rules for inferring the solution and provides necessary knowledge to assist the user to give an advice. Database stores the gating data for defining the gating dimensions. The procedure base stores the numerical or drawing programs that can be called upon during gating configuration. The control unit monitors the change on the working memory and controls the evolution of the gating configuration.

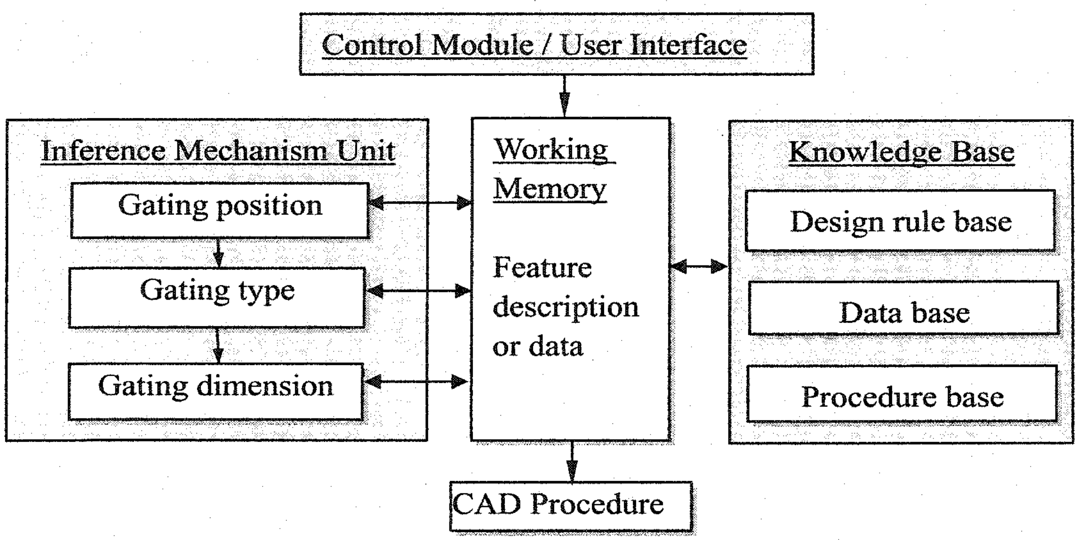


Fig 7.3 Architecture of feature-based gating CAD

7.2.3 Mechanism of feature-based gating CAD

As discussed earlier, the process of gating configuration is divided into three steps, that is, determination of gating position, selection of gating type and definition of gating dimension. The determination of gating position is based on the parting feature and the gating position rules. The parting features are classified into categories which are commonly used in die castings. After the gating position has been determined, the gating type is selected from predefined gating types. Then the required information for the definition of gating dimensions is acquired from the data base or the user by interactive design so that appropriate dimensions are defined. After the gating dimension has been defined, a complete gating configuration is obtained.

7.2.3.1 Determination of gating position

The determination of gating position depends on the parting feature, which includes the geometry and size of parting primitive as well as the parting location on the parting primitive. Parting primitive refers to the primitive which the parting line is situated at. Due to the diversity of parting features, it is difficult to build an abstract feature that adequately matches all features of parting primitives. A classification scheme is proposed for parting feature. According to the primitives defined in the Chapter 4 and common gating ceases, the parting primitives are classified into four basic categories: flat plate, sleeve, sphere crown and U-entity. Fig7.4 shows the classification scheme of parting primitives.

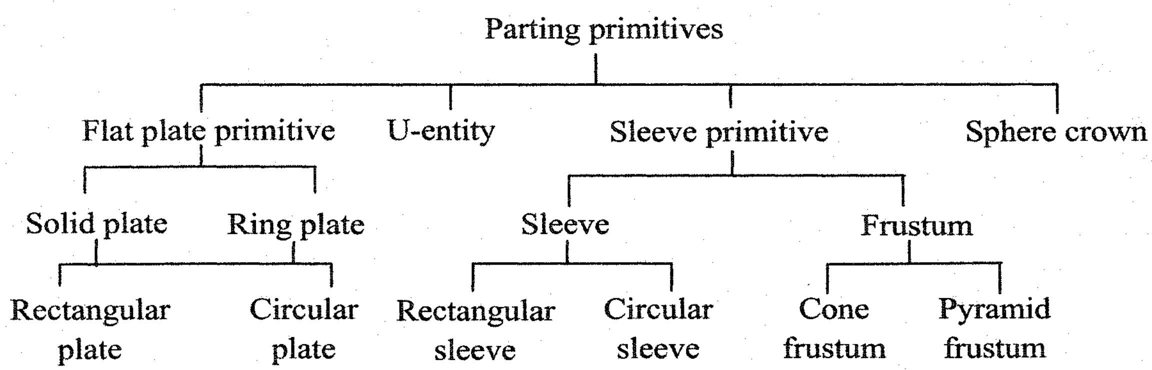


Fig.7.4 Classification of parting primitives

According to the gating design rule and practice, the gating positions for each parting primitive are defined as follows:

i) Flat plate parting primitive

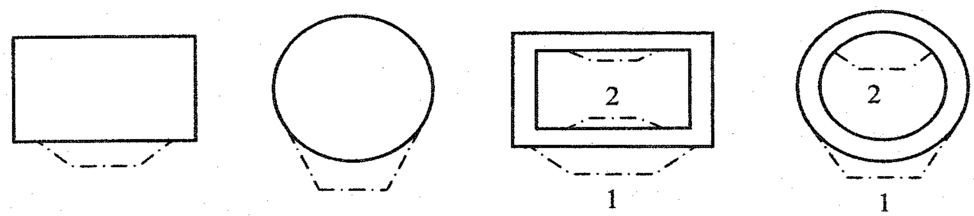
Flat plate parting primitives refer to those parting planes that are located on the large surface of a flat plate. There are two cases in this category:

Solid plate

The gating position of this category is recommended to be located at the side edge of the plate. If the plate is rectangular, the parting position should be on the long side edge, as shown in Fig. 7.5 (a).

Ring plate

The gate should position either inside or outside the ring plate as shown in Fig. 7.5 (b).



(a) Solid plate parting plane (b) Ring plate parting plane

Fig. 7.5 Recommended gating position for plate parting plane

ii) Sleeve (frustum) parting plane

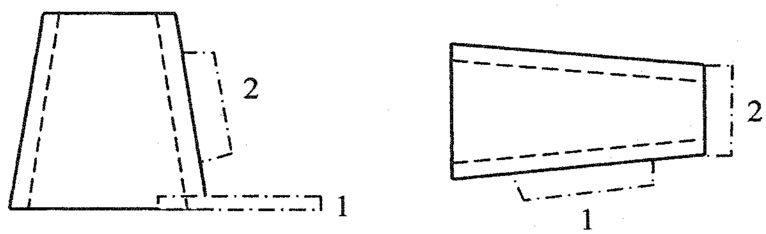
Sleeve parting plane refers to the parting plane that is located on a sleeve or frustum primitive. There are two cases in this category:

Parting plane at bottom face

The parting plane is located at the bottom face of a sleeve or frustum primitive. There are two gating positions to be recommended in this case. The first gating position is at the bottom face of the sleeve, and the other is located at the generating line on the side surface. In this case, the orientation of the sleeve is vertical.

Parting plane through axis

In this case, the parting direction is perpendicular to the axis of the sleeve. There are two gating positions to be recommended in this case. The first gating position is at the end face of the sleeve, and the other is located at the generating line of the side surface. In this case, the orientation of the sleeve is horizontal. The recommended gating position for the sleeve parting plane is shown in Fig.7.6.



(a) Parting plane at bottom face (b) Parting plane through axis

Fig.7.6 Recommended gating position for sleeve parting plane

In general, the gating position at the generating line is not recommended for rectangular sleeve and pyramid frustum primitive.

iii) Sphere crown

In die casting, sphere crown is a common primitive. In general, the parting plane is located along the bottom periphery of the sphere crown, as shown in Fig. 7.7(a), or through the symmetric axis, as shown in Fig. 7.7(b). The gating position is recommended as shown in Fig. 7.7.

iv) U parting entity

U parting entity refers to parting plane that involves an unsealed frame. In this case, symmetric gating position will be recommended, i.e., the parting position is located in the middle symmetric area, as shown in Fig. 7.8.

(a) Parting plane at bottom face (b) Parting plane through axis

Fig.7.7 Recommended gating position for sphere crown

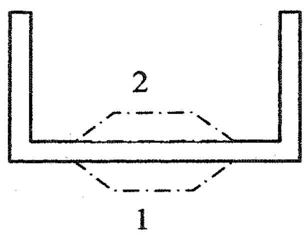


Fig. 7.8 Recommended gating position for U parting plane

Fig. 7.9 gives the inference graph of the gating position. By defining the categories of parting plane, the gating position can be inferred with production rules. For example, if a cone frustum is the parting primitive, the gating position should be inferred as follows:

Rule 01: IF parting primitive = cone frustum
AND parting plane = bottom face
AND height:diameter $\geq R_{crt}$
THEN gating position = generating line
gating position = bottom edge

Rule 02: IF parting primitive = cone frustum
AND parting plane = bottom face
AND height:diameter $< R_{crt}$
THEN gating position = bottom edge

The parting feature in the date section is transferred from the stage of parting scheme design and has the following format:

Parting Feature:

Parting primitive = cone frustum

parting plane = bottom face

Diameter = <diameter>

Height=<height>

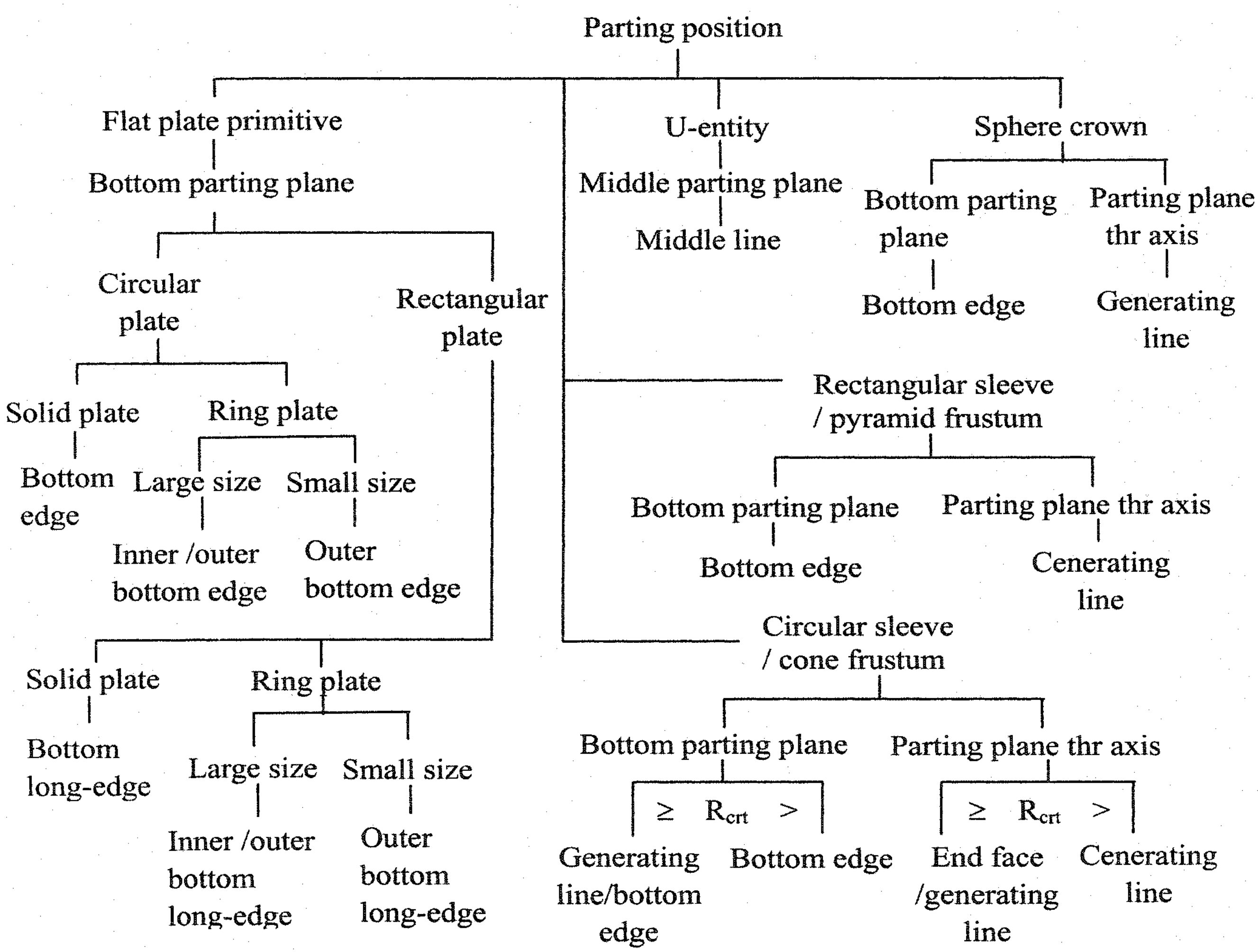
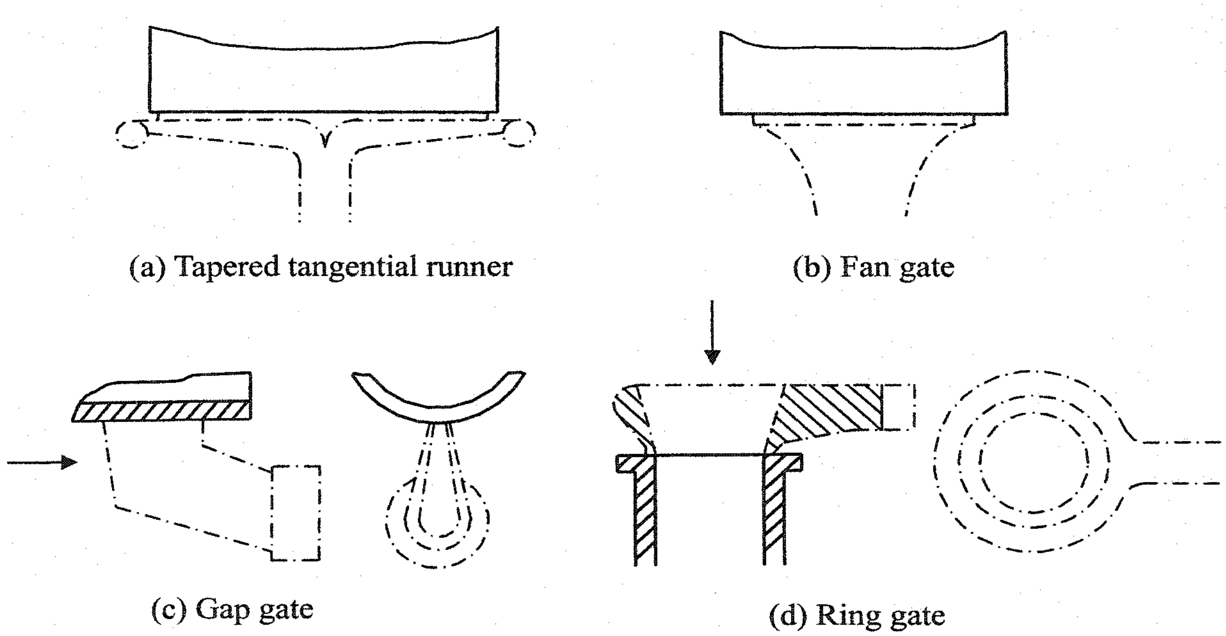


Fig.7.9 Gating position inference graph

After a gating position is determined, an appropriate procedure will be invoked to mark the positing in the primitive and stored in the working memory. At present, only single primitives are involved in the knowledge-based gating design. However, it would not be difficult to extend the analysis to multiple parting primitives and second level primitive (adjacent to the parting primitive).

7.2.3.2 Selection of gating type

The gating type includes the selection of the type of gating system and the manner that the ingate connects to the cavity. In the present study, four types of gating system are included. These are fan gate, tapered tangential runner, ring gate and gap gate. The four types of gating system are shown in Fig. 7.10. For a tapered tangential gate and fan gate, there are two connecting manners between the ingate and the cavity, i.e., edge connection and overlap connection. The two configurations are shown in Fig. 7.11.



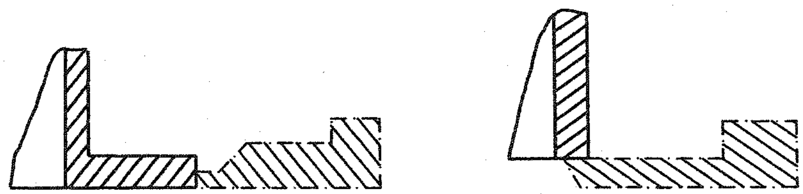
(a) Tapered tangential runner

(b) Fan gate

(c) Gap gate

(d) Ring gate

Fig. 7.10 Four types of gating system



(a) Edge connection (b) Overlap connection

Fig. 7.11 Two different connections

The selection of gating type follows the general rules as below:

i) Fan gate is suitable for castings with thick wall, narrow ingate and long filling distance casting;

ii) Tapered tangential gate is preferred for wide and thin wall casting;

iii) Gap gate is required for sleeve casting with deep cavity. The parting plane is located at the bottom face of the sleeve casting. The use of a gap gate would depend on the casting geometry and the position of parting plane.

iv) Ring gate is required for long sleeve casting. The parting plane is located in a axial plane of the sleeve casting. The use of a ring gate would depend on the casting geometry and the position of the parting plane.

The gating type is mainly dependent on the gating position on the parting primitive. Fig.7.12 gives the gating type inference graph. In the selection of gating type, heuristic knowledge is required.

The selection of the ingate connection is based on the follows rules:

i) For plate parting feature, edge connection is selected;

ii) For sleeve parting feature, overlap connection is selected.

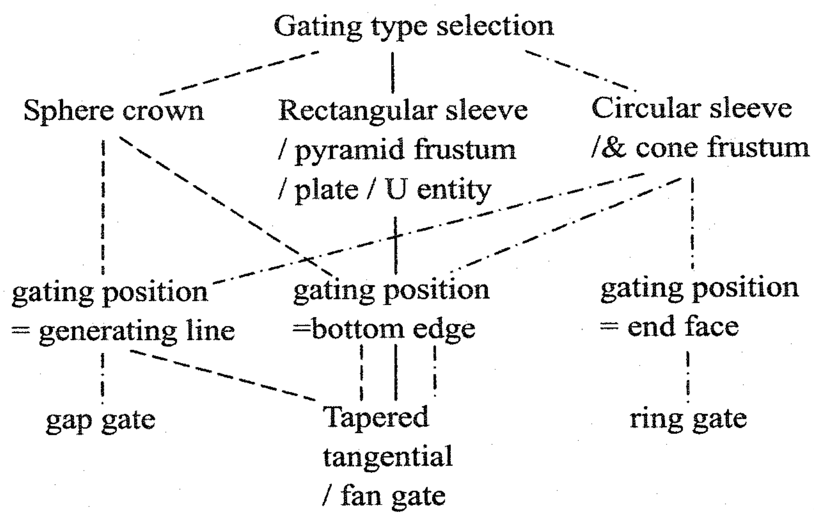


Fig.7.12 Gating type inference graph

The configuration of the connection is shown in Fig. 7.13.

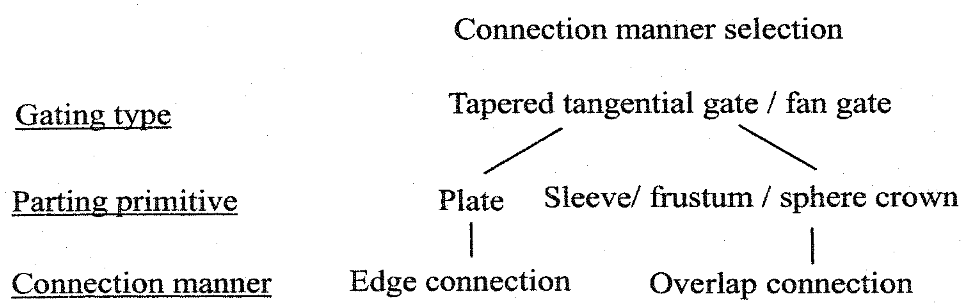


Fig. 7.13 Selection of connection type

7.2.3.3 Determination of gating dimensions

Once the gating position and the gating type have been determined, the gating dimensions can be calculated. The gating system includes three elements, i.e., ingate, runner and sprue. The determination of gating dimensions is to define sizes of each gating element.

Since the work focuses mainly on cold chamber die casting, the sprue design is not involved in the present research. An overall rule for the determination of gating size is that the cross-section area over gating system should continuously decrease from the biscuit to the ingate. This rule prevents flow turbulence and eddies in the gating system [Wilsom 2001]. There are two phases in the determination of gating dimension, i) cross-section shape, and ii) cross-section area of gating elements. A proportional method and a backward strategy are introduced in the determination of gating dimension. According to the overall rule for the determination of gating size, the ingate area should be kept minimal in the gating system. The idea of the proportional method is to determine the area of each element of the gating system by a number of ratios based on the ingate area, and the backward strategy is that the gating configuration begins from the ingate element backward to the sprue step by step. The cross sectional shape of gating elements is standardized. Such treatment is suitable for the knowledge-based design of gating system. As basic data of gating configuration, the determination of ingate dimension is discussed below:

i) Determination of ingate sizes

In general, the cross-section of the ingate is supposed to be rectangular. Two dimensions are involved: ingate width and depth. Traditionally, the cross-sectional area of the ingate can be calculated by following formula:

$$A_g = V/(\nu_g\ t) \qquad (7\text{-}1)$$

Where A_g: ingate area

V: casting volume

v_g: ingate velocity

t: filling time

The variables in the above formula are usually considered separately. Its shortcoming is that the effect of the die casting machine is not taken into account. In recent years, P-Q^2 technique has been widely used in determining gating parameters. The P-Q^2 diagram shows the inter-relationship between gating system of a die and injection system of a die casting machine. Using P-Q^2 diagram, the metal flow rate through the runner and the gate can be predicted for a given set of machine conditions. In this section, the gating area is determined based on the P-Q^2 method according to filling parameters calculated in the fuzzy evaluation stage. In such a way, the parameter selection can be correlated with the characteristics of a given machine. A typical P-Q^2 diagram for a specific machine (J1125t) [Zhang et al 1997] is shown in Fig.7.14, from which a particular gate area can be selected for a specific metal flow rate. The long dot line, short dot line and dot dashed line in the figure indicate the results of change in injection pressure, plunger velocity and plunger diameter. The P-Q^2 algorithm has been widely described [Rao et al 1989, Dan 1980], and will not be discussed in detail here.

After the cross-sectional area is obtained, the width and the depth sizes can be balanced. The determination of gate dimension is as follows:

i) Gate depth should be greater than 0.25 mm;

ii) Gate depth should be less than the wall thickness of adjacent casting in the gate area;

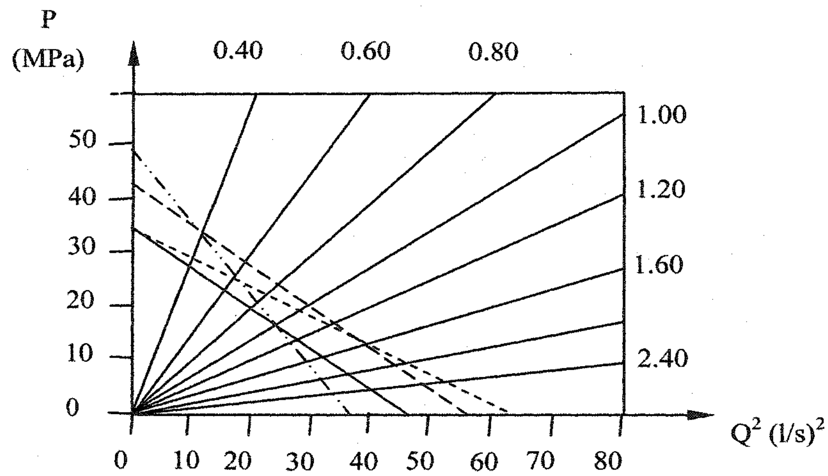


Fig.7.14 A typical P-Q^2 diagram for a specific machine (J1125t)

iii) Gate width should be less than half the perimeter of castings if the parting plane is elliptical or circular;

iv) Gate width should not be less than half edge length of the rectangular plate parting plane;

v) A comb gate should be used if an excessive thinner gate is presented;

vi) The length of ingate should be less than 5mm.

The ingate depth is one of the critical dimensions that has be established first. The selection of the gate depth should satisfy the requirements for atomization of the flow jet, trimming, and finishing. A well-developed atomization model of the molten metal flow stream is proposed [NADCA 1996]. The well-developed atomization of metal flow is considered to result in quality die castings. However, in most cases, the gate depth is selected from the data list based on other casting characteristics. In fact, the depth of the gate is closely related with wall thickness of adjacent casting. In the present study, an interpolation function is proposed to calculate the gate depth of Al alloy castings based on the casting thickness in the gate area. The function is as follows:

$$D_g = \frac{D_{g\max} - D_{g\min}}{h_{\max} - h_{\min}}(h - h_{g\min}) + D_{g\min} - kh \tag{7-2}$$

where

D_g: gate thickness (mm)

D_{gmax}: allowable maximum gate depth (mm) (D_{gmax}=3mm)

D_{gmin}: allowable minimum gate depth (mm) (D_{gmin}=0.25mm)

h_{max}: maximum casting thickness in gate area (mm) (h_{max}=6mm)

h_{min}: minimum casting thickness in gate area (mm) (h_{min}=0.5mm)

h: casting thickness in gate area (mm), $h_{max} \geq h \geq h_{min}$

k: coefficient, k=0 for normal case; k=0.2-0.4 for satisfying the requirements for atomized flow jet or the requirement of high surface finish.

The values in the parentheses are recommended ones. For the more thinner or more thicker wall, D_g is prescribed as follows:

$$\begin{cases} D_g = h & \text{for } h < h_{min} \\ h = h_{max} \quad \text{in Equation (7-2)} & \text{for } h > h_{max} \end{cases} \tag{7-3}$$

The thickness of 0.5-6.0mm covers most cases of casting wall thickness. If the wall thickness is below 0.5 mm, the depth of gate takes the equivalent value of the wall thickness. If the wall thickness is over 6 mm, the depth of gate can take the allowable maximum gate depth. After the gate depth is determined, the width can be obtained by the following formula:

$$W_g=A_g/D_g \tag{7-4}$$

Where

W_g: width of the gate, mm

A_g: area of the gate, mm^2

D_g: depth of the gate, mm

If tapered tangential runner or gate width priority is selected, the width of the gate should be determined prior to its depth. In this research, a gate width less than the diameter of castings and greater than sixty percent of the diameter is recommended for elliptical or circular parting plane and gate width equal to the edge length or greater than seventy percent of edge length is recommended for rectangular plate parting plane. In this case, the gate width can be defined through a ratio R:

$$R=\frac{\text{gate width } W_g}{\text{gating edge length } L_c}$$

The following values are suggestion for R:

Minimum value: 0.7

Maximum value: 1.0

Then the following formulas are used to define gate size.

$$\begin{cases} W_g=R*L_c \\ D_g=A_g/W_g \end{cases} \tag{7-5}$$

Lc is a characteristic length, it is the perimeter of a sleeve for a ring gate, and the height of a sleeve for a gap gate. The determination of ingate dimension is shown in Fig. 7.15.

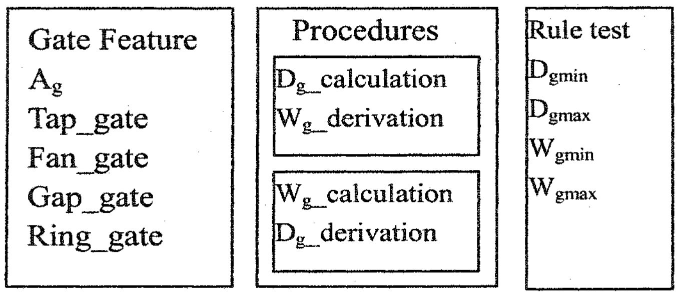


Fig. 7.15 Determination of ingate size

Generally, the determination of the ingate dimensions is divided into four steps:

Step 1: Select ingate area according to flow rate,

Step 2: Calculate width and depth,

Step 3: Verify ingate rules, and

Step 4: Adjust width and depth.

ii) Determination of runner sizes

A runner is a metal passage segment between the sprue and the ingate. As described earlier, an overall rule for determining runner size is that the cross-sectional area over runner should continuously decrease from the biscuit to the ingate, and a proportion method is used to define the cross sectional area of each gating element. Based on the cross section area, the sizes dimensions can be determined by dependency relationship of dimensions. In this section, A tapered tangential runner and a fan gate are discussed.

Tapered tangential runner dimension

In general, a tapered tangential runner has the following basic components: two branches of tapered runner (RB), a delta region (DE), two shock absorbers (SA), runner extension segment (RE) and main runner (RM). Fig. 7. 16 shows a typical runner configuration with element cross sectional shapes, critical position and dimensions of a tapered tangential system.

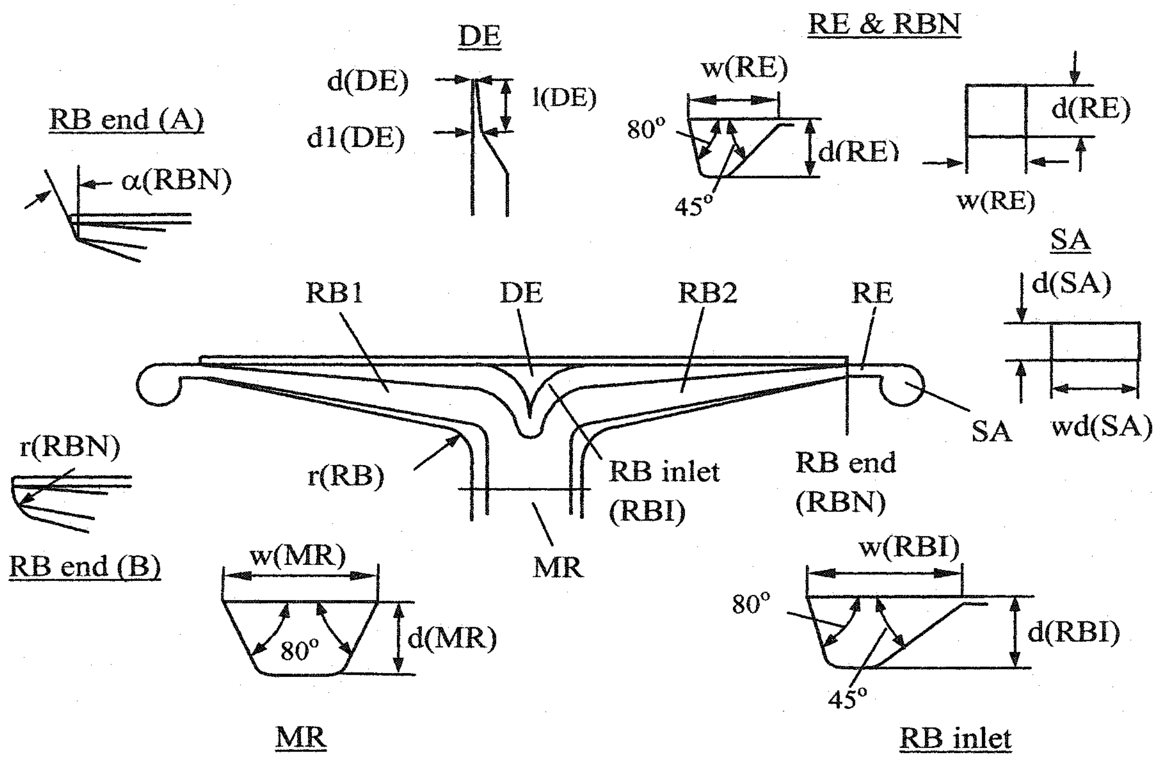


Fig. 7.16 Tapered tangential runner

The dimension dependency relation or quantitative relation among the elements in the tapered tangential gating system is shown in Fig. 7.17. In the dependency relation network, there are two types of nodes: dimension nodes and function nodes. Each dimension node is associated with a dimension value. Each function of node is linked with one or several input dimension nodes and one output dimension node. When a dimension value is changed, the functions that use this dimension as input node are then activated to update this change to the output dimension nodes. This change propagation process is carried out continuously until no dimension value change is required. The relation functions are given in Fig. 7.18.

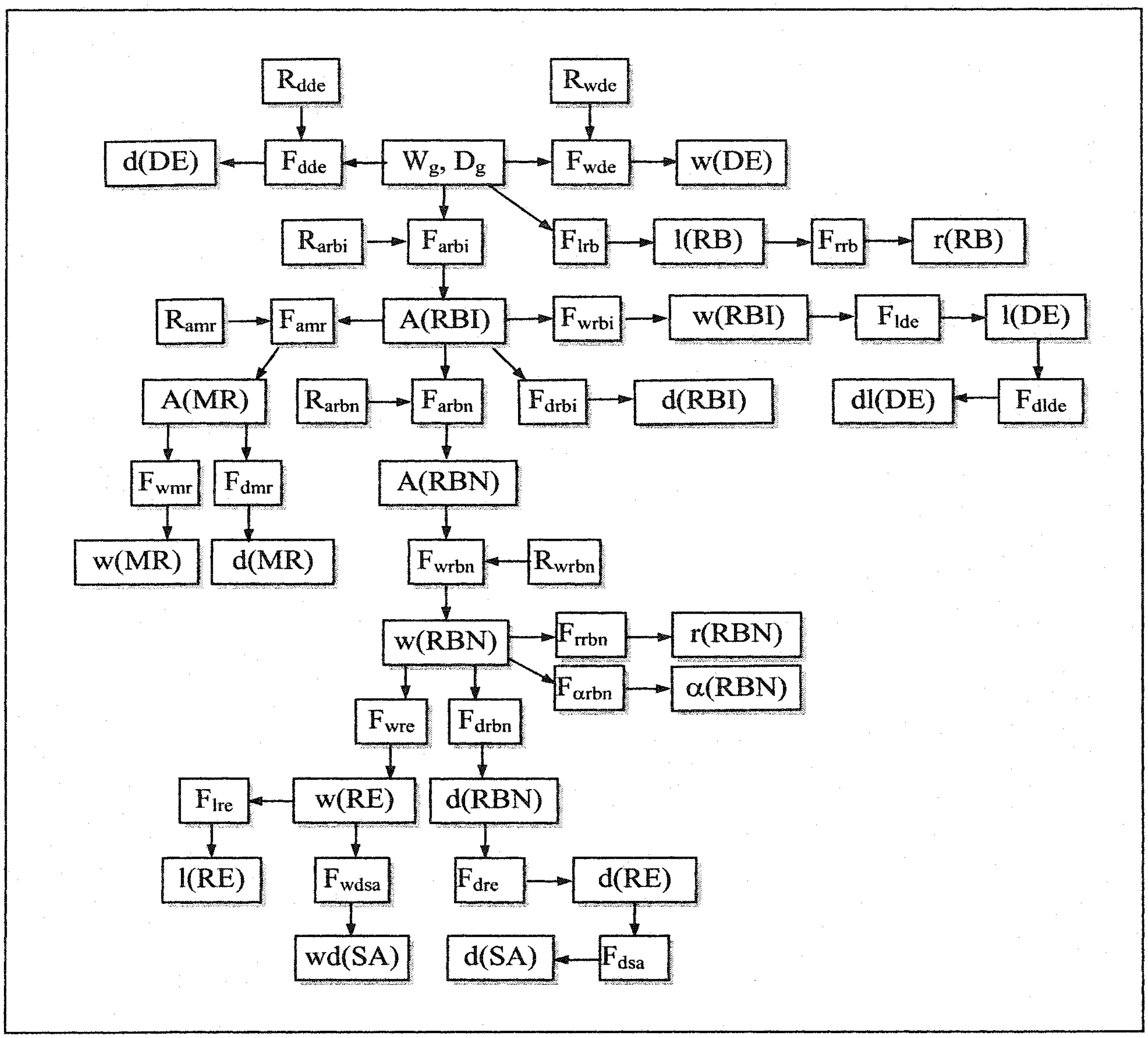


Fig. 7.17 The dimension dependency network in tapered tangential system

R is specific ratio between two elements. R_{dde} is the ratio of depth between the delta gate and the ingate:

$$R_{dde} = \frac{\text{delta gate depth d(DE)}}{\text{gate depth } D_g}$$

F_{dde}: $d(DE)=R_{dde}*D_g$	F_{dlde}: $dl(DE)=\tan(15^\circ)*l(DE)+d(DE)$
F_{wde}: $w(DE)=R_{wde}*W_g$	F_{arbn}: $A(RBN)=R_{arbn}*A(RBI)$
F_{lrb}: $l(RB)=W_g$	F_{wrbn}: $w(RBN)=(R_{wrbn}*A(RBN))^{1/2}$
F_{rrb}: $r(RB)=0.2*l(RB)/2$	F_{drbn}: $d(RBN)= A(RBN)/ w(RBN)$
F_{arbi}: $A(RBI)=R_{arbi}*(W_g*D_g)$	F_{dre}: $d(RE)= d(RBN)$
F_{amr}: $A(MR)=R_{amr}*A(RBI)$	F_{wre}: $w(RE)= w(RBN)$
F_{wmr}: $w(MR)=1.595\ A(MR)^{1/2}$	F_{dsa}: $d(SA)= d(RE)$
F_{dmr}: $d(MR)= 0.678\ A(MR)^{1/2}$	F_{wdsa}: $wd(SA)=3* w(RE)$
F_{wrbi}: $w(RBI)= 1.727*(A(RBI)/2)^{1/2}$	F_{lre}: $l(RE)=4* w(RE)$
F_{drbi}: $d(RBI)=0.794*(A(RBI)/2)^{1/2}$	F_{rrbn}: $r(RBN)=2* w(RE)$
F_{lde}: $l(DE)= w(RBI)/2$	$F_{\alpha rbn}$: $\alpha(RBN)=45^\circ$

Fig. 7.18 Relation functions used in the dimension dependency network of tapered tangential runner

The following values are suggested for R_{dde}:

Minimum value: 0.0

Maximum value: 1.0

The recommended value is 1.0.

R_{wde} is the ratio of width between the delta gate and the ingate:

$$R_{wde}=\frac{\text{delta gate width } w(DE)}{\text{gate width } W_g}$$

The following values are suggested for R_{wde}:

Minimum value: 0.0

Maximum value: 0.3

The recommended value is 0.25.

R_{arbi} is the ratio of cross-section area between the RB inlet area and the gate area:

$$R_{arbi} = \frac{\text{RB inlet A(RBI)}}{\text{gate area } A_g}$$

The following values are suggested for R_{arbi}:

Minimum value: 1.0

Maximum value: 3.0

The recommended value is 1.5.

R_{arbn} is the ratio of cross-section area between the runner terminal and the runner inlet.

$$R_{arbn} = \frac{\text{RB end area A(RBN)}}{\text{RB inlet area A(RBI)}}$$

The following values are suggestion for R_{arbn}:

Minimum value: 0.1

Maximum value: 1.0

The recommended value is 0.5.

R_{amr} is the ratio of cross-section area between the main runner and the branch runner inlet area:

$$R_{amr} = \frac{\text{MR area A(RM)}}{\text{BN inlet area A(BNI)}}$$

The following values are suggestion for R_{amr}:

Minimum value: 1.0

Maximum value: 1.8

The recommended value is 1.1.

R_{wrbn} is the ratio between the width and the depth of runner end:

$$R_{wrbn} = \frac{\text{RB end width w(RBN)}}{\text{RB end depth d(RBN)}}$$

The following values are suggestion for R_{wrbn}:

Minimum value: 1.0

Maximum value: 2.0

The recommended value is 1.0.

Fan runner dimension

There are two types of fan runner: linear width and linear depth, as shown in Fig. 7.19. The dimension dependency network is show in Fig.7.20, and the relation functions of the fan gate are shown in Fig. 7.21.

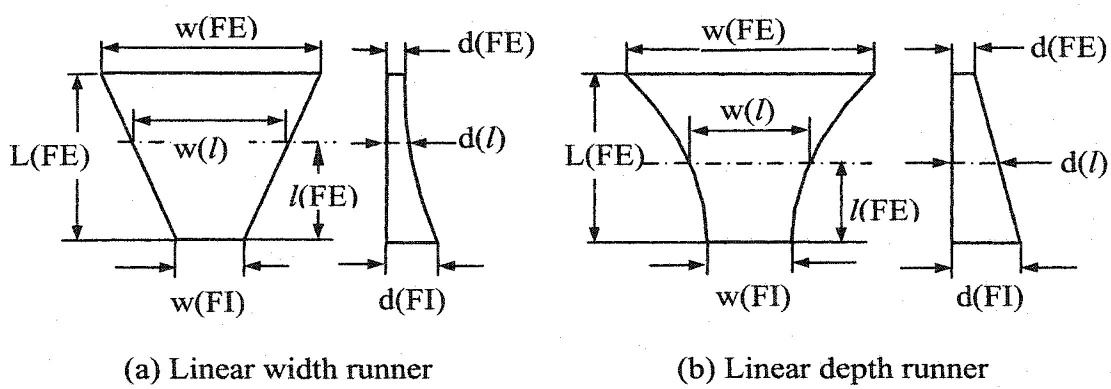


Fig 7.19 Two different fan runners

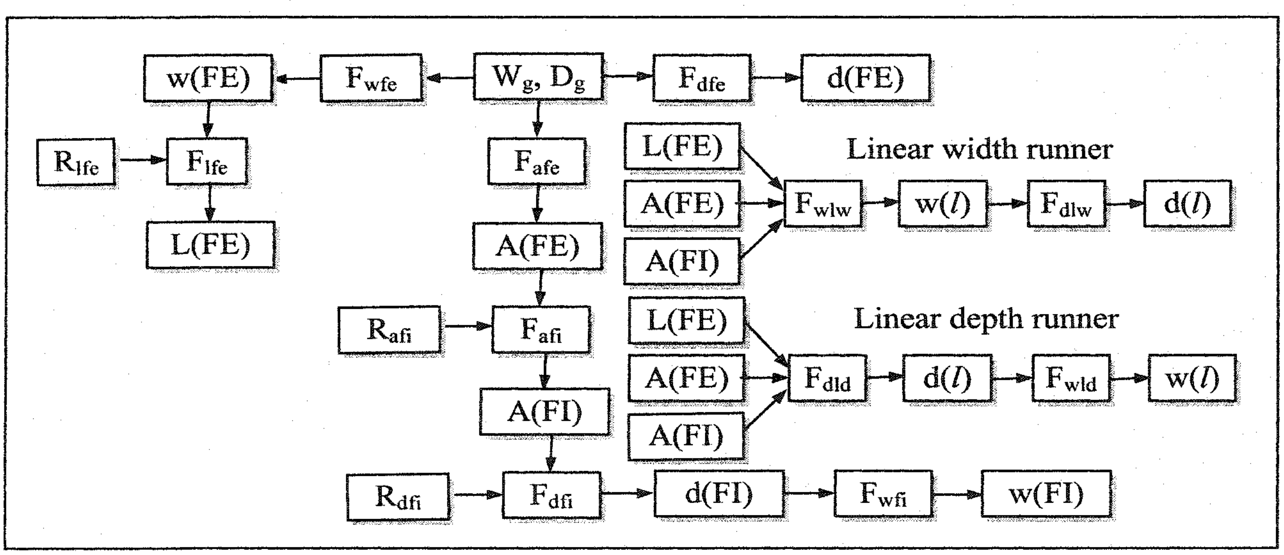


Fig. 7.20 The dimension dependency network of fan gate

F_{wfe}: $w(FE)=W_g$
F_{dfe}: $d(FE)=D_g$
F_{afe}: $A(FE)= w(FE)*d(FE)$
F_{afi}: $A(FI)=R_{afi}* A(FE)$
F_{dfi}: $d(FI)=(A(FI)/ R_{dfi})^{1/2}$
F_{wfi}: $w(FI)= A(FI)/d(FI)$
F_{lfe}: $L(FE)=R_{lfe}* w(FE)$
F_{wlw}: $w(l)=(w(FE)-w(FI))*(l(FE)/L(FE))+w(FI)$
F_{dlw}: $d(l)=((w(FE)*d(FE))+(R_{afi}-1)*(w(FE)*d(FE))*(1-l(FE)/L(FE))/w(l)$
F_{wld}: $d(l)=(d(FE)-d(FI))*(l(FE)/L(FE))+d(FI)$
F_{dld}: $w(l)=((w(FE)*d(FE))+(R_{afi}-1)*(w(FE)*d(FE))*(1-l(FE)/L(FE))/d(l)$

Fig. 7.21 Relation functions used in the dimension dependency network of fan gate

R_{afe} is the ratio of cross-section area between the runner exit and the inlet:

$$R_{afe}=\frac{\text{runner inlet area } A(FI)}{\text{runner exit area } A(FE)}$$

The following values are suggestion for R_{afe}:

Minimum value: 1.2

Maximum value: 2.0

Recommended value is 1.5.

R_{dfi} is the ratio between the width and the depth of runner inlet:

$$R_{dfi}=\frac{\text{runner inlet width } w(FI)}{\text{runner inlet depth } d(FI)}$$

The following values are suggestion for R_{dfi}:

Minimum value: 1.0

Maximum value:3.0

Recommended value is 1.67.

R_{lfe} is the ratio between the width of runner exit and the overall length of runner:

$$R_{lfe} = \frac{\text{runner length L(FE)}}{\text{runner exit width w(FE)}}$$

The following values are suggestion for R_{lfe}:

Minimum value: 0.5

Maximum value:1.5

Recommended value is 1.0.

The dependency network describes the quantitative relation among gating elements by a number of functions. The gating dimensions can be determined from a dependency network. The user is only required to choose a series of proportions (ratios) between related elements. During the determination of ratios, the system can provide heuristic knowledge which can help the user to choose a reasonable ratio. The heuristic knowledge is stored in the knowledge base in predefined text. For example, the following knowledge text is given for the selection of R_{lfe}, the ratio of width of runner exit and the overall length of fan runner, as shown in Fig. 7.22.

Selection R:	HK Base: RunnerLength_GateWidth (R_{lfe})
RunnerLength_GateWidth: R_{lfe} ...	1. Low value results in large flow angle, the flow in perpendicular direction weakens. 2. High value increases gating volume, and the flow in perpendicular direction strengthens. Range: 0.5-1.5 Recommended Value: 1.0

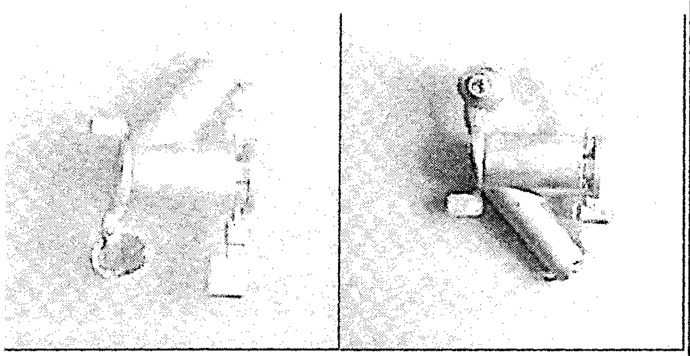
Fig.7.22 The knowledge text for selection of R_{lfe}

7.3 Case-based gating design

There are several benefits of applying case-based technique in the determination of gating configuration. It allows the designer to propose solution quickly and efficiently, and it allows the designer to address a successful solution provided that the new case is similar to an existing case in the case library. Case library stores numbers of successful cases which can be referenced for a new case. The case-based approach has been widely used in solving engineering problems [Kwong and Smith 1998, Wang 1992]. A case-based mechanism consists of three basic components, a case library, a case based inference unit and library-editing unit. In this study, the case library is used to store successful cases for gating configuration. A rich library is critical for the case-based system. The inference unit determines which case will be extracted from the library and it influences the work efficiency of the system. The library-editing unit is used to access library and to edit cases, such as addition, removal and adaptation of the case. There are four important issues in the development of a case-based system, these are case library structure, case indexing, case retrieval and case adaptation. In this subsection, the four issues in gating case-based system are addressed.

7.3.1 Case library structure

The case library is composed of a number of cases which are in a predefined structure and organization. In order to obtain a complete solution, any case in the library should include two parts at least. The first is problem description, and the second is a problem solution. The problem description refers to an expression of problem characteristics related with the problem-solving, and the solution gives the corresponding answer to a specific problem. In gating design, the problem description is the characteristic depiction of a die casting, including geometry, dimension, wall thickness, material. A typical description for a tap body is shown in Fig. 7.23. The casting description is in the form of predefined text or a casting drawing or a casting picture. The description provides the geometry and main dimension information for comparison with new cases. If a new case is matched with an existing one in the case library, the gating system is applicable or is available for reference to the new case. Then the solution will be given. The solution yields the gating configuration, including gating position, gating type and gating dimension. Fig. 7.24 shows the gating configuration for the tap body, and basic information of the gating system. Since the cases stored in the case library are collected from real production and prove to be successful, they are reliable. Although two cases are seldom identical, the solution devotes a valuable reference provided there is a similarity between the two cases.



Casting case

Casting description
Casting name: tap body
Material: Zn alloy
Reference size:
Main tube: diameter 54 mm
length 82
wall thickness 2 mm
Second tube: diameter 32 mm
length 90mm
wall thickness 2 mm
Cut angle: 45°

Fig. 7.23 The case description of a tap body casting

Gating solution
Gating position: end face
Gating type: ring gate
Gating size:
Ingate: thickness; 0.5 mm
Ingate width: 170 mm
Ingate length: 1mm
Main runner: width 15 mm
thickness 8.5 mm
length 20 mm
Ring runner: entrance width 12 mm
entrance thickness 8.5 mm
end width 6 mm
end thickness 1.5 mm
Sprue: diameter 38 mm
height 58 mm
Die casting Machine: 150 t (hot chamber)
Filling velocity: 58m/s

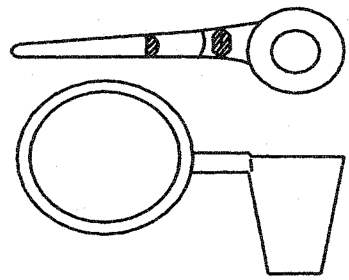


Fig. 7.24 The solution of gating configuration for tap body

A case stored in the case library includes two segments of description and solution. A case is an independent unit. All cases are organized in hierarchy of class of cases. The diagram of a basic class of plate parting feature is organized in a tree hierarchy as show in Fig. 7.25.

Plate parting feature
- Circular plate
  - Circular sleeve
  - Rectangular sleeve
  - ...
- Rectangular plate
  - Circular sleeve
  - Rectangular sleeve
  - ...
- ...

Fig. 7.25 Diagram of a basic class of plate parting feature

7.3.2 Index and retrieval of cases

In general, there are large numbers of cases stored in the case library. The cases in the case library are organized in specific form, and a case may cover a category of castings. The position of all cases in the library should be registered so that the required case can be correctly and quickly retrieved. Therefore, indexing is necessary. Since parting feature is closely related with gating configuration, the parting feature is used to index the gating case. The index consists of two parts: first level feature and second level feature. The first level feature indicates the parting feature (PF), i.e., plate, vertical sleeve, horizontal sleeve and so on. The second level feature (AF) is an auxiliary part to indicate further necessary information, such as the feature of primitive adjacent to parting feature to retrieve a specific case or a category of cases. A case index has following format:

Indexing $\{PF, AF_1, AF_2, \cdots\}$

The index list is prescribed by the system builder and in a particular order.

Case retrieval requires a combination of search and matching. In case retrieval, similarity analysis between two cases is often used for the determination of the closest case [Kwong and Smith 1998] as useful and efficient approach. Since the description of a casting object requires the geometrical features including primitive shape, dimensions and topological relationship, the similarity function is difficult to be established. Therefore, a gradual case-approaching is adopted. As mentioned in the previous section, the case library is organized in a hierarchical structure of case class, and the gradual approaching method is to match the feature one after the other. Obviously, the parting feature is matched at first and a basic class is addressed. A basic class includes a number of subclasses, which will be retrieved if there is no following feature to be matched. If an additional feature is given, a further match will be performed. All cases in a subclass will be retrieved. If the features are given sufficiently, a closest case can be retrieved. The retrieval of gating case employs the production rule, for instance, the case of plate parting feature is retrieved as follows:

IF *plate parting feature* THEN *cases of plate class retrieved*

IF *plate parting feature & circular shape* THEN *cases of circular-plate subclass retrieved*

IF *plate parting feature & circular shape & circular sleeve on it* THEN *cases of subclass of circular sleeve on circular-plate retrieved*

The antecedent part is in prescribed format and the consequent part links up the case class.

7. 3.3 Case adaptation

In principle, there is no complete match between two cases in die casting. The retrieved solution cannot be used discriminately in general, and an adaptation is always required. The adaptation of gating solution is based on the heuristic method. After a gating solution is retrieved from the case library, the new case should be compared with the retrieved case by the designer. By comparison, adaptation should be determined in three phases, these are gating position, gating type and gating dimension. Since the gating cases stored in the case library are successful examples collected from die casting production, the gating solution is valuable for the gating design. If a retrieved case is useful for the new case, the gating position and gating type can remain unchanged. In most cases, only adjustment of gating dimensions is required. Anyway, the system always gives the heuristic adaptation rules if adaptation is required. For instance, for horizontal sleeve parting feature, the following heuristic rules are suggested:

Horizontal sleeve parting feature

? *Gating position adaptation*

end position (?description)

generating line position on parting line (?description)

? *Gating type adaptation*

ring gating type for end gating position (?description)

fan gate or tapered tangential runner for generating line position (?description)

? *Gating dimension adaptation*

ring gating dimension (?description)

fan gating dimension (?description)

tapered tangential gating dimension (?description)

The description can give more information on the calculation of dimension. The flow chart of case-based gating design is shown in Fig. 7.26. If there is no case to match the new one, the system will switch to the feature-based CAD branch.

7.4 Gating example with feature-based CAD approach

To illustrate the capability of the feature-based gating CAD, the casting example of a casing body used in parting design and parameter design is taken. From the parting scheme design stage, the parting scheme has been determined as shown in Fig.4. 19 (a). The parting

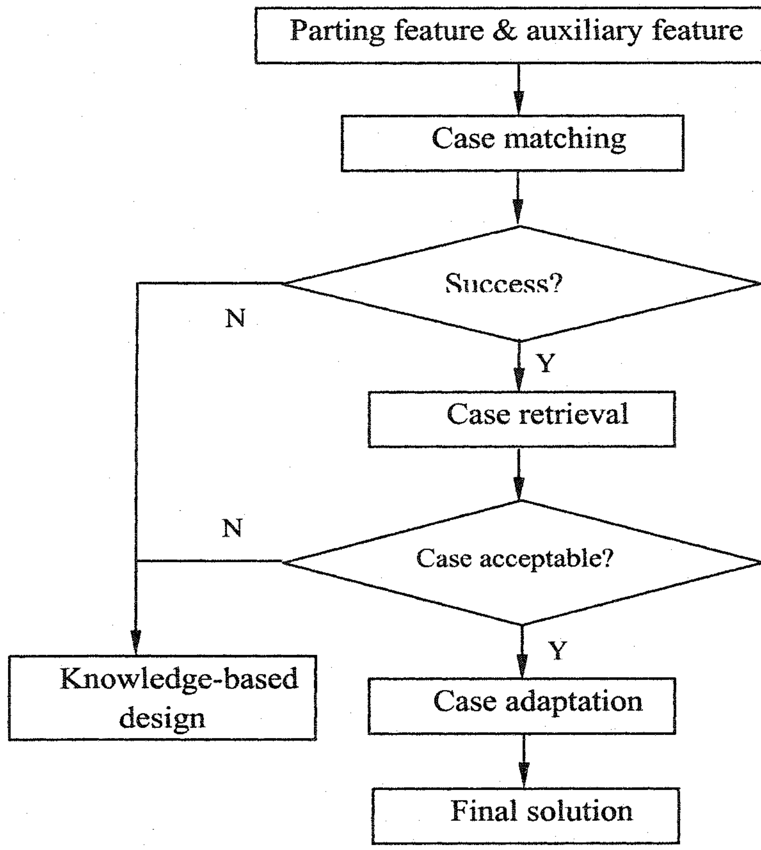


Fig. 7.26 The flow chart of case-based gating design

feature file transfers the parting feature to gating design.

Parting_primitive(vertical_ellipse_sleeve)

Sleeve Long_Diameter: 400mm

Short_Diameter: 200mm

Height: 25mm

Wall thickness:2.4mm

Orientation: (0,0,1)

Location: (0,0,0)

Parting_position: (ellipse_sleeve_bottom_face)

The design starts with reading out the parting feature of the casting, then infers the gating position. The gating position rules would be searched in the knowledge base. For vertical ellipse sleeve primitive, there are two rules to suggest the gating positions,

R01 IF *vertical ellipse sleeve parting feature & sleeve_height less than or equal to 0.8* short_ diameter*

THEN *bottom edge gating position at long edge*

R02 IF *vertical ellipse sleeve parting feature & sleeve height greater than 0.8*short diameter*

THEN 1) *generating line gating position at end side*

2) *bottom edge gating position at long edge*

Obviously, the first rule is matched for a given parting feature. The parting positions are located at the bottom long edge of the ellipse base. It should be noted that the gating position is inferred only based on the first level feature, i.e., primitive involved in the parting plane. The gating type is strongly related with the gating position. The knowledge base approach would make the following suggestion:

R01 IF *vertical sleeve parting feature & bottom edge parting position*

THEN *fan gate or tapered tangential runner*

R02 IF *vertical sleeve parting feature & generating line parting position*

THEN *gap gate*

Decision should then be made between the fan gate and the tapered tangential runner. This decision is conducted by heuristic knowledge. In general, the fan gate is chosen for short width of gate and with requirement of controlling filling direction. The tapered tangential runner is suitable for wide width of gate and short filling distance.

R01 IF gating edge long <150mm>

THEN tapered tangential runner

A value of 150 mm for gating edge length is predefined as the critical value. If the gating edge is longer than 150 mm, a tapered runner is suggested. For this casting example, the tapered tangential is selected. After the gating position and the gating type are selected, the next step is to determine the gating dimensions. In the knowledge base, the gating dimension configuration is represented as a knowledge object, and the gating dimensions can be determined once certain necessary data are acquired. Suppose that the casting will be produced on a die casting machine of F580c (Frech 580t_cold chamber). The characteristics of the machine are shown in Table 7.1 and the data from P-Q^2 analysis are shown in Table 7.2.

Table 7.1 Characteristics of machine F580c

Type of machine	Cold chamber
Max. metal pressure (MPa)	48
Max. flow rate (l/s)	31
Plunger diameter (mm)	70
Max. plunger velocity (m/s)	9
Fast-shot speed setting (%)	65
Discharge coefficient	0.65

Table 7.2 Data from P-Q^2 analysis of machine F580c

A_g (cm^2)	1.00	1.50	2.00	2.50	3.00
Q (l/s)	11.5	14.9	16.9	18.0	19.0
v_f (m/s)	112	95	79	73	64
t_f (ms)	60	48	41	37	35

The required parameters are given in Table 7.3. According to the required flow rate, the gate area A_g can be selected form the data of P-Q^2 analysis. Taking the flow rate into consideration, a gate area of 2.0 cm^2 is suitable. For a flow angle of 30°, the actual gate area is found to be 2.32 cm^2. The major dimensions of gating elements and cross section can be calculated and are shown in Table 7.4. The symbols in the table can be seen in Fig. 7. 16.

Table 7.3 Required parameters

Vol_{cast} (cm^3)	t_f (ms)	A_g (cm^2)	Q(l/s)	L_c (mm)	R	R_{arbi}	R_{wmr}	R_{amr}	R_4^*
584	38	2.32	15.37	400	0.85	3.0	1.67	1.1	0

* R value of 0 indicates without use of corresponding element.

Table 7.4 Major dimensions of gating elements

W_g (mm)	D_g (mm)	A(RBI) (mm^2)	d(RBI) (mm)	w(RBI) (mm)	A(MR) (cm^2)
340	0.70	350 (single)	15.00	33.00	7.70
d(MR) (mm)	w(MR) (mm)	A(RBN) (mm^2)	d(RBN) (mm)	w(RBN) (mm)	r(RBN) (mm)
18.00	45.00	278	13.00	28.00	56

According to the structure and the dimensions, the gating configuration on the casing body is shown in Fig. 7.27. Since the parting primitive is a vertical sleeve, the overlap connection is adopted.

Once the gating system has been configured, the design of die casting scheme is finished. A complete die casting scheme is provided including parting scheme, die casting parameters and gating configuration. With the generated die casting scheme, the die is fabricated and trial production is made. The result shows that the given casting scheme is reasonable. A sound die casting is produced as shown in Fig. 7.28.

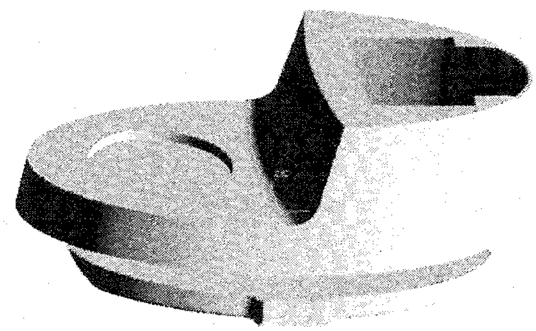


Fig. 7.27 Tapered tangential gate configuration

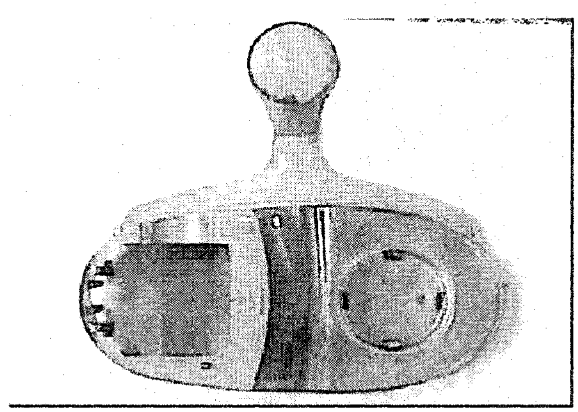


Fig.7.28 Casing body produced with the given die casting scheme

7.5 Verification and modification of the die casting scheme

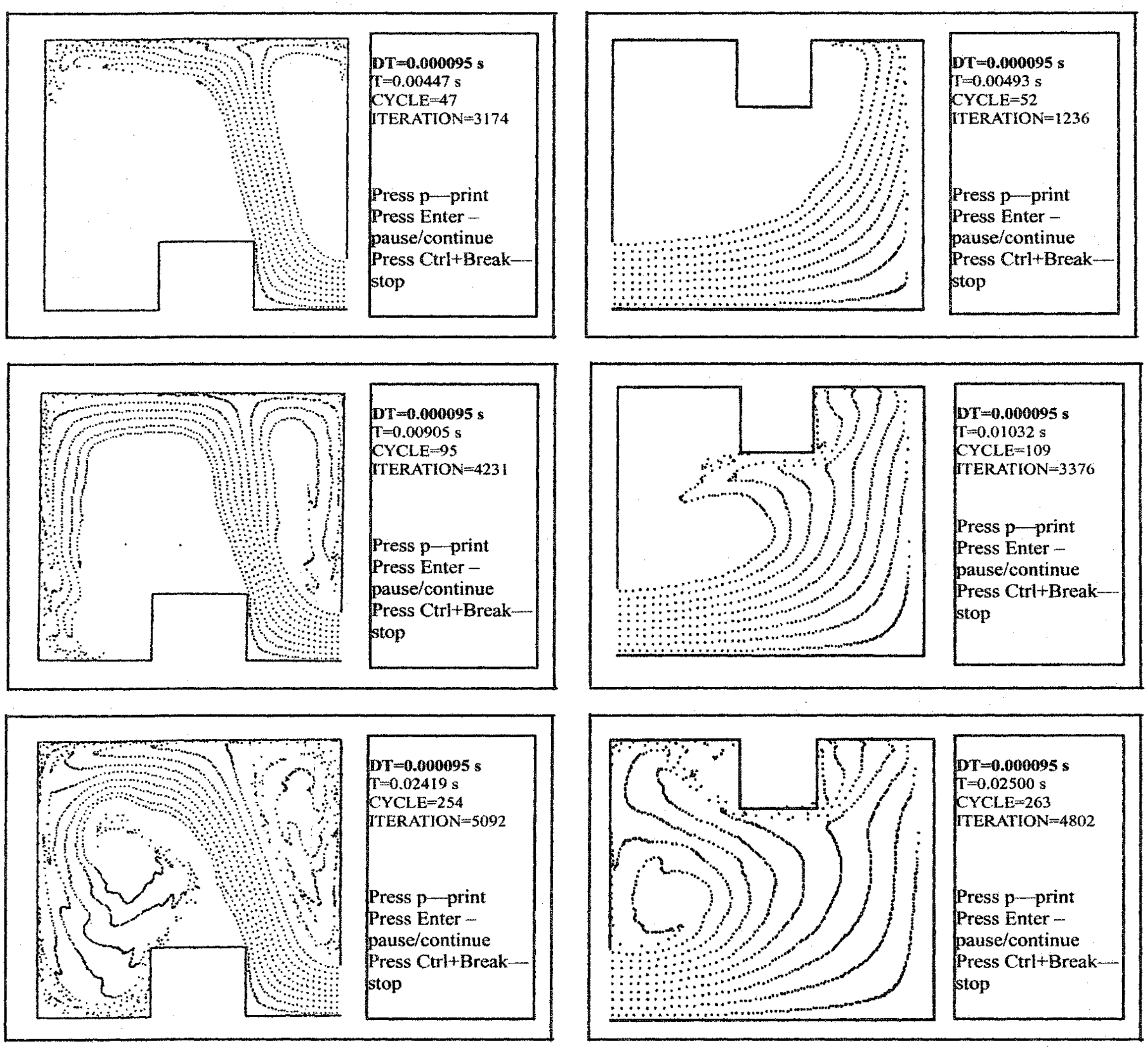
Since die casting scheme design is based on the empirical knowledge, design errors can occur, leading to casting defects. In this case, verification is required. Die modification is very expensive and time-consuming. Computer simulation is introduced to perform verification and modification of the die casting scheme.

7.5.1 Computer simulation

The purpose of using computer simulation package is to confirm whether there are filling problems in a chosen die casting scheme. In the earlier stage of the research, an initial effort is made on the DFLOW package by Lu and Lee [1993, 1999]. The DFLOW package was developed for the simulation of cavity-filling of thin-wall die castings. This package can predict the cavity-filling patterns for different filling parameters and in different gating configurations. Application of the package in this work is intended to demonstrate the effect of varying filling parameters and gating configurations, and then to show the ability of the simulation package to verify filling parameters and gating configurations.

The DFLOW was developed based on the finite difference method (FDM). DFLOW solves the continuity equation and momentum equations by an iterative procedure. The basic idea of the algorithm comes from the vorticity transport equation. Based on the vorticity definition, the Navier-stokes equations can be transformed into the vorticity equation. The vorticity is independent of the pressure, which means that the vorticity field is independent of the pressure field. For this reason, any field of pressure inserted into the Navier-Stokes equations will assure that the resulting velocity field carries correct vorticity. If the tentative velocity field is altered by the addition of the gradient of an appropriate potential function, the resulting velocity field will carry the correct vorticity, and has vanishing divergence. Accordingly this velocity field will be uniquely determined. This is the essence of the DFLOW algorithm for the finite difference solution.

Fig. 7.29 shows the simulation results for different gating positions. The flow pattern shown in Fig. 7.29 (a) corresponds to a gating position against an obstacle, and Fig. 7.29 (b) corresponds to a gating position far from a obstacle. In the first case, back pressure can be generated by entrapped air, air bubble or porosity may occur in the closed region. A model of back pressure in a die cavity has been established by Lee and Lu [1999]. Comparing the two cases, the second one is to be preferred.



(a) Ingate against an obstacle (b) Ingate far from an obstacle

Fig.7.29 Filling patterns with different ingate positions

A commercial simulation software called FLOW3D can also be used to simulate the filling process, which integrates the flow and heat transfer. The software uses VOF algorithm and is powerful for flow simulation. Fig. 7.30 shows the simulation results of a casing body with a die casting scheme generated from Chapter 5.

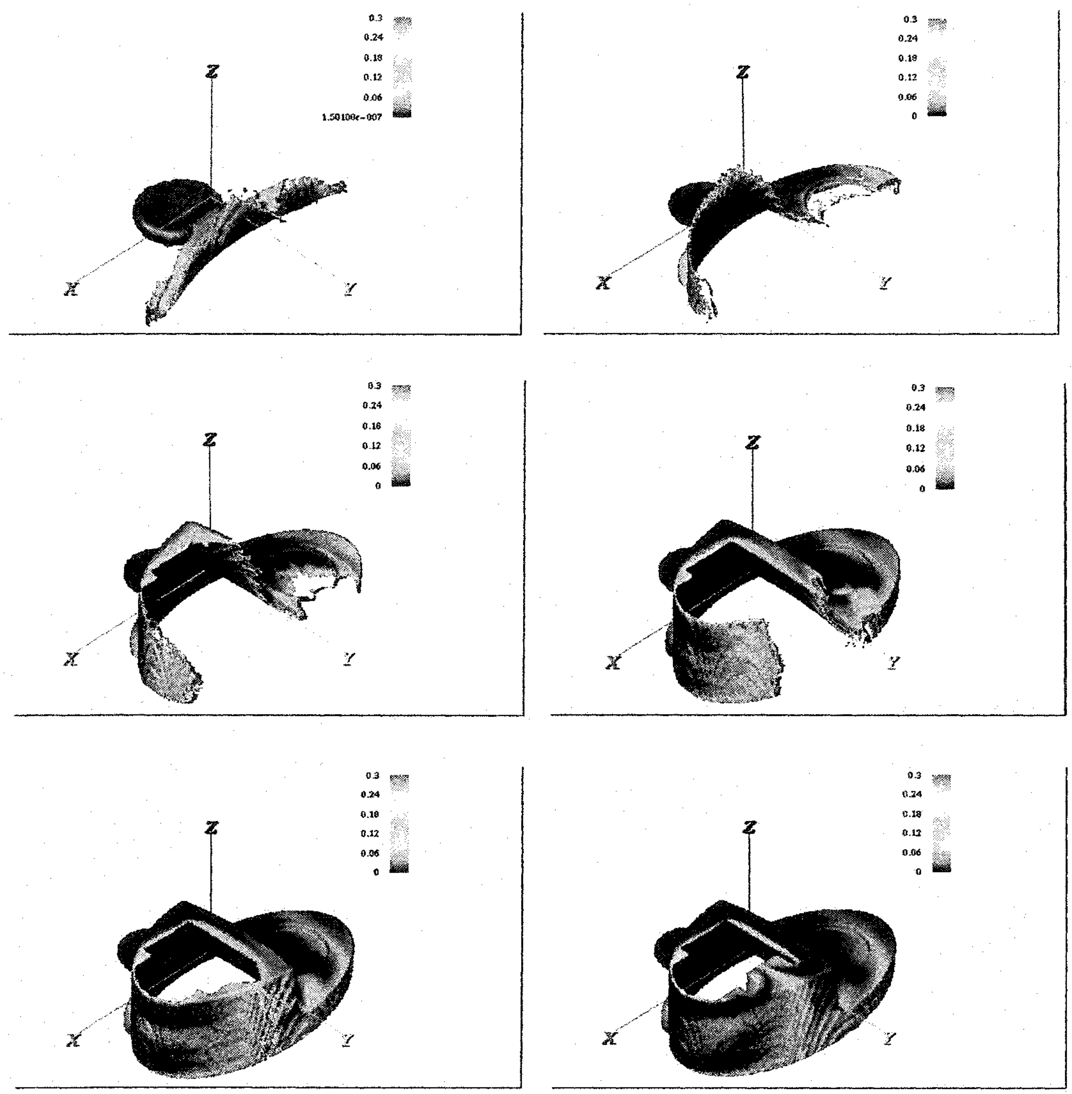


Fig. 7.30 Simulation result of filling process of casing body

The computer simulation can provide a visual observation of the filling pattern. The filling behavior in different gating configurations, different parting schemes and different casting parameters can be examined. Different die casting schemes can be tested by computer simulation.

Traditionally, a basic way to find casting defects is by trial production, which is most effective, but time-consuming and expensive. With the development of the computer simulation and numerical technique, the simulation of casting processes is used to predict the casting defects. Since this method can reduce expensive trial production, it has attracted the attention of die casters. The simulation is a useful tool to verify die casting scheme design.

7.5.2 Modification of the die casting scheme

By simulation or trial production, if an unfavorable filling pattern or casting defect is predicted, modification of the die casting scheme is then needed. The principal purpose of die casting scheme design is to ensure sound castings in die casting production. However, the design process involves many fields of engineering and science, such as heat transfer, fluid dynamics, alloy materials, filling theory and process control. No design is guaranteed to be perfect. Design errors are not usually explicitly revealed. Manifestation of design deviations is shown by casting defects. A die casting scheme is considered to be in error if an unsound casting is produced. The judgment of the design deviation is based on the defect characteristics. According to the judgment of casting defects, modification of die casting scheme can be made. Once the defects are specified, the design errors can be analyzed and design modification should be made. A defect can be caused by multiple factors, there is no one to one correspondence relationship between a casting defect and a design error. Resolution of such problems requires experience and expert knowledge [Wei 1985, Ceng and Xiao 1993, Roshan and Jain 1990, Liang and Liu 2000, Liu 2000, Luo and Zhou 2001, Lu and Lee 2001]. Therefore an expert system approach can be employed to analyze defects and perform design modification.

In any modification system, causal ordering procedure is used to analyze causal dependencies and produces a graphical relationship that encodes these dependencies. The causal dependency graph defines a complete error set for a particular defect. Fuzzy inference approach is used to identify parameter design errors based on the fuzzy analysis of parameter variables and influence strength of errors to find out the possibility of each parameter error for a specific defect. By fuzzy inference, the parameter errors can be identified and the sequence of modification actions can be determined for elimination of the parameter errors. In addition, for gating and casting operation errors, modification actions are sorted and the sequence of the modification is based on the difficult degree to take these actions. In addition, a self-adapting mechanism is introduced to improve the validity of the correlative relationship between the parameter and the defect. Based on the modification results, the fuzzy relationship matrix is then updated and recorded. The self-adapting mechanism can make fuzzy relationship more reasonable after the system is used for a period of time.

The computer simulation and modification of die casting scheme are expected as an auxiliary tool to improve the validity of the design of the die casting scheme, and can be incorporated into the present design system.

7.6 Summary

In this chapter, the following works have been completed:

i) A gating design system is developed based on the knowledge-based technique. The case-based approach and feature-based CAD approach are combined to generate a gating configuration. The case-based approach is used to match the successful cases of gating configuration, and feature-based approach is used to design a gating system under interactive CAD environment based on the parting feature and gating knowledge.

ii) A knowledge representation scheme based on the knowledge object is developed. A dimension dependency network is developed to define gating dimensions. Based on the object-oriented approach, the system framework integrates the qualitative and quantitative analysis of gating design. With the proposed approach, gating position, the gating type and gating dimensions can be recommended.

iii) Successful gating cases have been collected from die casting production. The cases are illustrated in the form of figures and text which give detailed gating configuration. Moreover, a case library structure, case retrieval and case adaptation are discussed.

iv) A design example is illustrated based on the feature-based approach. The example has been successfully produced with the recommended die casting scheme.

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

In this chapter, the research of the hybrid design support system for a die casting scheme (HDDS) is summarized. An overall discussion, major contributions and research finding are addressed. Some possible enhancements and extensions for the current design system are given.

8.1 Overall discussion and contributions

The design of a die casting scheme often determines the performance of a die casting die. There are three major tasks in die casting scheme design: parting scheme design, parameter selection and gating configuration. There are a number of critical issues to be solved in order to design successfully a die casting support system. In parting scheme design, the recognition of concave region features and calculation of release direction are important issues. The previous work is limited to geometrical features composed of polyhedral or planar surfaces, and the optimal parting direction is determined by minimizing the number of undercuts. In fact, parts with planar surfaces are few and the optimal parting direction is related with multiple factors. In the present research, a 3D ray detection method is employed for the recognition and extraction of concave features from a die casting. With the ray detection method, each concave feature is assigned a block factor. A set of concave regions and their hulls can be built by examining the block factor. The points in a concave hull carry geometrical information required for the determination of normal vector and release direction. The minimal release direction principle and the vector plane approach are proposed to solve for the release direction of any arbitrary concave feature. The proposed approach extends the concave feature analysis to curved planes by approximating the surfaces with a series of discrete face pieces or points. The accuracy of the algorithm varies with the size of the face pieces. In principle, with the vector plane and extreme vector approach, the release direction of arbitrary curved surfaces can be solved. Discrete surface is a feasible approach for analyzing release direction problem of curved plane. The parting direction and parting position is selected based on the release direction and geometrical feature of parting primitive. The optimal parting scheme is determined by an evaluation and optimization procedure. The determination of optimal parting direction does not depend upon minimizing the number of undercuts only.

The conventional way to select a die casting parameter is to find a particular value from a published chart, diagram or use of simple formula based on certain characteristics of a die casting. In such a way, there may be considerable fluctuation in selecting die casting parameters due to a wide data range and difference in the experience of individual designers. In order to overcome the above drawbacks, a fuzzy synthetic evaluation technique has been proposed for the determination of die casting parameters. With fuzzy synthetic evaluation, the factors influencing parameter object are divided into levels, and evaluation on the parameter object is performed against levels. The fuzziness of factors can be well processed and the effect of each factor can be synthetically considered. The fuzzy synthetic evaluation does not restrict the number and characteristic of factors involved.

This approach is suitable for the determination of design parameters. Compared with conventional ways, the fuzzy evaluation can provide consistent and reasonable resolution. The work on fuzzy synthetic evaluation attempts to propose a new way for the selection of die casting parameters.

In the development of a gating design system, a combination of case-based approach and feature-based CAD approach is introduced. The feature-based CAD approach can provide an interactive gating design under the guidance of design rules. The case-based approach can provide successful instances for reference. This is one of the most efficient ways especially for experience-dependent design problems. If the knowledge and cases are sufficiently collected, a correct design of gating system can be assured. In addition, the excessive dependence on the experience of individual designer can be alleviated by such a design system and the design can be performed by designers with different technical backgrounds. In case-based reasoning gating design, use of a similarity model can improve case retrieval.

The class of casting parts involved has been limited to those that have conventional two-plate dies and rectilinear removal of the die casting in which the most die castings can be produced. In addition, the HDSS thus developed is a prototype only. However, the scope of the system can be extended and improved to handle more design consideration and geometrical primitives. The major contributions of this research can be summarized as follows:

i) An extreme vector concept and vector plane approach are developed to extend the release direction analysis from planar surface to curved surfaces. This is an important extension for feature analysis;

ii) An principal candidate parting direction is proposed for the first time to solve blind matching problem in die casting scheme generation;

iii) The application of fuzzy synthetic evaluation overcomes the drawbacks of the conventional method in determination of the die casting parameters. It presents a new way in the parameter design for die casting;

iv) A knowledge-based approach is used to improve the CAD system in gating design. It is one of the first of its kind and contributes to the automation of the computer-aided die design.

8.2 Summary of research

The hybrid support design system developed for die casting processes in this thesis covers the essential tasks of die casting scheme design. This offers an assistant tool for improving die design technology. Parting scheme design, parameter determination and gating configuration are three important issues in die design. In this dissertation, a systematic research on the methodology of a computerized system for the three issues has been made. With the methodology proposed, a hybrid design support system is developed. The work in this study provides a strong technological support towards the automation of die design process. As a result of the study, the following conclusions can be drawn:

i) Parting scheme design is dependent on the geometrical feature of die castings. The present study analyzes the geometrical features influencing parting scheme design, and a 3-D ray detection method is proposed to recognize and extract the parting feature of concave regions in a die casting. With this method, a concave region in an arbitrary geometry and orientation can be detected.

ii) The determination of release direction of a concave region is crucial for the design of a parting scheme. A general mechanism for the release direction analysis is proposed based on the extreme vector and vector plane approach, the release direction of both planar and curved surface can be analyzed.

iii) According to the characteristics of a parting scheme design, a feature-based approach is presented for generation of the parting scheme. An efficient matching mechanism is proposed for parting scheme generation based on the concept of the principal candidate parting direction. It is found the design efficiency is increased and all feasible parting schemes can be generated

iv) The system is capable of evaluating parting schemes. Rule-based evaluation models are developed by formulizing design rules into computable test or satisfaction functions, the quantitative evaluation on the performance of parting schemes is realized. With the evaluation results, an optimal scheme can be determined by a multiple-criteria optimization approach.

v) A fuzzy synthetic evaluation approach is introduced for the determination of the value of process parameters. The approach can quantitatively and synthetically handle multiple fuzzy factors and provide reasonable and consistent results. The application of fuzzy evaluation presents a new method in the selection of process parameters.

vi) A gating design system is developed based on the knowledge-based technique. A combination of case-based approach and feature-based CAD approach is proposed for the recommendation of gating configuration. The case-based approach can provide successful instances for reference, and the feature-based CAD approach can design a gating system based on the casting feature and design knowledge. The approach is efficient to improve conventional CAD system in die casting scheme design.

vii) The proposed system has been initially tested based on real castings from industry and the recommended schemes were found to be capable of producing sound and acceptable castings. The present work can provide a support towards the automation of die design process.

8.3 Suggestion for further research

The hybrid design support system for die casting scheme is an important step in the development of design automation of die casting die. There are still many areas requiring improvement and enhancement for further work.

i) The present analysis is only for the removal of a casting in rectilinear motion, a rotational removal should be added to further research

ii) At present, there are many commercial graphic softwares and simulation softwares on the market. A link or communication with such softwares will increase the capability of the proposed system. Further work should be needed to link up propriety software and general software with a common platform.

iii) The design process in proposed system is still not fully automatic, further development needs to include more intelligent techniques to automate the die design process

iv) It is needed to develop a similarity analysis model for efficient case retrieval in case-based reasoning

v) The effectiveness and the usefulness of the HDSS system depend on the completeness and the integrity of the design knowledge. More design knowledge and geometrical primitives should be added to increase its application. In addition, virtual manufacturing technique is gaining the attention of researchers and enterprises in casting industry, a virtual prototyping system for die design can be carried out based on the present work.

REFERENCES

Akira Nishmura and Tatsuichi Fukusako, "A method of calculating cavity filling time and its evaluation in die casting", *Foundry*, Vol. 57, No.8, p.503-509 (1985)

Allsop D.F., D. Kennedy, *Pressure Diecasting*, Part 2, Pergamon Press, Oxford (1983)

Andresen W.T., "Metlflow: Computer aids for die casting", *Die Casting Engineer*, No.5, p.38-43 (1986)

Andresen W. T., "Computer simulation and analysis of liquid metal flow and thermal conditions in die casting dies", *SDCE 14th International Die Casting Congress and Exposition*, G-T87-019, May 11-14, Toronto Ontario Canada (1987)

Andrew Kusiak and Juite Wang, "A knowledge-based system for building scheduling Models", *Artificial Intelligence in Optimal Design and Manufacturing*, Prentice-Hall, Inc., pp173-197 (1994)

Anthory, A. Amsden and fracis H. Harlow, "The SMAC method: A numerical technique for calculating incompressible fluid flow", *Los Alamos Scientific Laboratory Report LA-4370* (1970)

Anzai Koichi, Toshio Uchida and Katsutoshi Kataoka, "prediction of defects in die castings by quisi three-dimensional flow analysis", *Foundry*, Vol. 60, No.12, p.763-769 (1988)

Balachandran M., Knowledge-Based Optimum Design, *Computational Mechanics Publications*, Hobbs the Printers Ltd (1993)

M.R. Barone and D.A. Caulk, "dieCAS- Thermal analysis software for die casting: modeling approach", *Die Casting Technology*, NADCA, October 18-21, Cleveland T93-013, p.19-25 (1993)

Barton H.K., *Product Design for Die Casting*, Society of Die Casting Engineers, River Grove, ILL. (1981)

Benedek Von Takach, "Some aspects of feed design for pressure die casting dies", *10th SDCE International Die Casting Exposition & Congress*, March 19-22, St. Louis, Missouri, Paper No. G-T79-094 (1979)

Ceng Yidan and Xiao Keze, "Expert system for analysis of casting porosity defects", *Special Casting and Nonferrous Alloys*, No.1, p.25-27 (1993)

Chen Jincheng, *Die Casting Technology Basis*, Beijing Radio Equipment Factory (1978)

Chen Lin-Lin, Visibility algorithms for mould and die design, Ph.D. Dissertation, The University of Michigan (1992)

Chen Lin-Lin, Shou-Yan Chou and Tony C Woo, "Parting directions for mould and die design", *Computer-Aided Design*, Vol. 25, No.12, p.763-767 (1993)

Chen C.W., C.R. Li and T.H. Han, "Numerical simulation of filling pattern for an industrial die casting and its comparison with the defects distribution of an actual casting", *AFS Transactions*, 94-154, p.139-146 (1994)

Chvatal, V., "A greedy heuristic for the set covering problem", *Mathematics of Operations Research*, 4(3), pp233-235 (1979)

Dan Zabel, "P-Q^2 diagram and its application" *Die Casting Engineering*, Vol.24, No.5, p.24-27 (1980)

Davey, K. and Hinduja, S., "The importance of process and model parameters in the pressure die casting process", *Proceedings of 5th International Manufacturing Conference*, Beijing, 2-4 April, p.345-349 (1991)

David Anthony, "Knowledge-based injection mold design automation", Thesis of Ph.D., University of Lowell (1990)

Dennis Earl Ferguson, "An Expert System for Injection Molding", Thesis of Ph.D., The University of Minnesota (1989)

Draper A.B. and I.W. Perkert, "An analytical approach to die casting", *Proceedings of 2nd National Die Casting Congress*, Detroit (1962)

Earle W. Rearwin, "Gating of aluminum diecastings", *Foundry*, November, p.106-113 (1960)

Editor group, *Handbook for Diecasting Die Design*, Mechanical Industry Press, Beijing (1981)

Er A., E. T. Sweeney and V. Kondic, "Knowledge-based system for casting process selection", *AFS Transactions*, 96-175, p363-370 (1996)

Fischer G.W., G.T. Ramaseshadri, "Parting line placement by decomposing part geometry into primitive features", *AFS transactions*, 97-12, p.359-366 (1997)

Frayce D., J.F. Hetu and C. A. Loong, "Numerical modeling of filling and solidification in die casting", *North American Die Casting Association*, Cleveland T93-011, Oct. 18-21, Cleveland, Ohio (1993)

Fu M.W., J.Y.H.Fuh, A.Y.C. Nee, "Undercut feature recognition in an injection mold design system", *Computer-Aided Design*, 31, pp777-790 (1999)

Ganter M.A., L.L. Tuss, "Computer-aided parting line development for cast pattern production", *AFS transactions*, 90-137, p.795-799 (1990)

Gary F. Ruff., "Metal casting virtual reality and strategies for growth", *Modern Casting*, (8) p.62-65 (1998)

George R. Doyle, Willard D. Kaiser &Thomas P. Groeneveld, “Computer-aided die design for zinc die casting dies”, *Die Casting Engineer*, No. 6, p.24-29 (1981)

George Salloum, Denis Charland and Alain Vinet, ”Castfeed: Interactive design of feeding systems for die casting molds”, *14th SDCE International Die Casting Exposition & Congress*, May 11-14, Toronto, Paper No. G-T87-094 (1987)

Graham E., “Shape feature modeling”, *Solid Modeling by Computers*, p.121-137 (1986)

Douglas K. Gross, “zinc die casting gate and runner design”, *Proceedings of the First China International Die Casting Congress*, 14-17 April, 1997, Beijing, p.139-147 (1997)

Group of Handbook, *Handbook of Mathematics*, People’s Education Press (1979)

Guleyupoglu S. et al, “Computer-aided orientation of castings in sand molds”, *AFS Transactions*, 94-111, p.529-535 (1994)

Guleyupoglu S. and J.L. Hill, “Draw direction and parting plane selection criteria for sand castings”, *AFS Transactions*, 95-117, p.259-264 (1995)

Guleyupoglu S., “Casting process design guidelines”, *AFS Transactions*, 97-83, p.869-876 (1997)

Kiran C. Gupta, *Classical Mechanics of Particles and Rigid Bodies*, John Press (1988)

Han Chuanliang, “Calculation of withdrawal force in die casting” *Nonferrous Alloys and Special Casting*, No.4 p.43-44 (1985)

Harlow, F. H., Shannon, J.P. and Daly, B.J., “The MAC method-A computing technique for solving viscous, incompressible, transient fluid-flow problems involving free surface”, Los *Alamos Scientific Laboratory Report* LA-3425 (1966)

Herman, E. A., Gating and system design now a simple science”, *Die Casting Engineer*, December, pp18-20 (1973)

Herman E.A., *Die Casting Dies: Designing*, Society of Die Casting Engineers, River Grove, ILL (1985)

Hill J.L., J.T. Berry, “Geometric feature extraction for knowledge-based design of rigging systems for light alloy castings”, *Fifth Engineering Foundation Conference on the Modlong of Casting, Welding and advanced Solidification Processing*, September 16-20, Davos, Switzerland (1990)

J.l. Hill, J.T. Berry and S. Guleyupoglu, “Knowledge-based design of rigging systems for light alloy castings”, *AFS Transactions*, 91-62, p.91-96 (1991)

Hiroyuki Nomura, Kazuhiko Terashima and Koji Keishima, “Prediction of die casting flow behavior by three dimensional simulation", *Foundry*, Vol.63, No.5, p.425-430 (1991)

Hirt, C.W., Nichols B.D. and Romero, N.C., "SOLA- A numerical solution algorithm for transient fluid flows", *Los Alamos Scientific Laboratory Report* LA-5852 (1970)

Hirt, C.W. and B.D. Nichols, "Volume of fluid method for the dynamics of free boundaries", *Journal of Computational Physics*, 39, p.201-225 (1981)

Hu Jhyfang, "Towards a knowledge based design support environment for design automation and performance evaluation", Dissertation of Ph.D., The University of Arizona (1989)

Hwang W.S. and Stoehr, R.A., "Computer simulation for the filling of castings", *AFS Transactios*, 87-141, p.425-430 (1987)

Hui K.C. and S.T.Tan, "Mould design with sweep operations-a heuristic search approach", *Computer-Aided Design*, Vol.24, No.2, Feb, p.81-91 (1992)

Hwang W.S., "Computer aided fluid flow analysis of the filling of casting system", PhD Thesis, university of Pittsburgh (1985)

Itamura M., N. Yamamoto, E. Niyama and K. Anzai, "Application of the flow visualization technique and flow simulation to die casting flow analysis", *Int. J. Cast Metal Res.*, No.9, p.139-143 (1996)

Jan Tzy-Cherng, Expert system for injection molding engineering thermoplastics, Ph.D. Dissertation, New Jersey Institute of Technology (1992)

Ji Shouxun, Yang Guozhu, Ding Wanli et al, "A CAD system for diecasting gating system based solid modeling", *Proceedings of the First China International Die Casting Congress*, 14-17 April, 1997, Beijing, p.391-395 (1997)

Jin Yuchun, Tian Guanghui, Xu Hong and Cheng Jun, "A CAD software system of die casting dies with personal computer", *Proceedings of the First China International Die Casting Congress*, 14-17 April, 1997, Beijing, p.386-390 (1997)

S.H. Jong, H.Y. Chou and C.R. Li, "Application of mold filling analysis in the design of die casting die", *Foundry*, Vol. 73, p.1-9 (1993)

Kai Tang, Congruent partition and certain computational geometry problems, Ph.D. Dissertation, The University of Michigan (1990)

Kaiser W.D., J.D. Sanders and P.D. Frost, "Production alternatives for making large thin-wall zinc die castings", *Die Casting Engineer*, No. 4, p.18-22 (1971)

Kalien L.H., C. Sturm and F. Odor, "Optimization of casting process through simulation for aluminum castings", *AFS transactions*, 92-162, p.1067-1077 (1992)

Kappel V.V., and G. Smalloum, "Modelling of the metal flow in the die casting process", *15th die casting congress & exposition*, Oct. 16-19 m1989, St. Louis, MO, Paper No. G-T89-041, p.1-4 (1989)

Koichi Anzai and Toshio Uchida, "Extended quasi three dimensional mold filling simulation method for die casting", *Foundry*, Vol. 62, No.2, p.90-95 (1990)

Kondic V., A. Er, E.T. Sweeney, "A knowledge-based system approach to the selection of casting methods", *Proceedings of Sixth International Mg. Conference with China*, Vol.2, p.273-278 (1993)

Kopf J., "Optimum value determination od basic die casting injection variables", *5th International Die Casting Congress*, Detroit, Paper No. 23, p.132 (1968)

Kotschi R.M., "The miss algorithms to fully computerize gating, risering and CAM tooling manufacture, part 1", *AFS Transactions*, 89-136, p.689-694 (1989)

Kulkarni A. and G.A. Stone, "Casting defect analysis expert system", *AFS Transactions* 100, p.881-886 (1992)

Kuo J-H, W-S Hwang, "Flow pattern simulation of metals in the shot sleeve during injection of die casting", *AFS Transactions*, 98-63, p.497-503 (1998)

Kwong K.C. and G.F. Smith, "Computational system for process design of injection moulding: Combining blackboard-based expert system and case-based reasoning approach", *The International Journal of Advanced Manufacturing Technology*, 14, pp239-246 (1998)

Lee K., C.T. Smith, "Computer-aided pattern design for casting process", *Proceedings of the 14th Conference on Production Research and Technology*, Ann Arbor, Michigan, p.365-371 (1987)

Lee W.B. and Chan S.K., "Computer-aided gating design for pressure diecasting dies", *Proceedings of the Advanced Manufacturing Technology Conference*, March 10-11, Hong Kong, p.332-330 (1989)

Lee E. Stanley, Qing Zhu, *Fuzzy and Evidence Resoning*, Physica-Verlag, A Springer-Verlag Company (1995)

Lee W.B. and H.Y. Lu, "Modeling of air back pressure in diecasting dies", *Journal of Materials Processing Technology*, Vol.91, No.1-3, 30, June, p.264-269 (1999)

Li Chuanshi, *Foundry Engineer Handbook*, Mechanical Industry Press, Beijing (1997)
Liang Weizhong, Liu Jun, "Establishment of inference engine of a quality assurance expert system for 462 cylindrier castings", *Foundry*, No.2, p.96-98 (2000)

Lin J.C. and C.C. Tai, " Runner optimization design of a die casting die", T*he International Journal of Advanced Manufacturing Technology*, p.14:133-145 (1998)

Lindsey D.M. and J.F. Wallace, "Heat and fluid flow in die casting process", *5th SDCE National Die Casting Congress*, Detroit, Paper No.12 (1968)

Lindsey D.M. and J.F. Wallace, "Effect of vent size and design, lubrication practice, metal degassing, die texturing and filling of shot sleeve on die casting soundness", *7th SDCE National Die Casting Congress*, Detroit, Paper No.132 (1972)

Liou Shun-Yuan, Diecast: A knowledge based approach to die casting design, Ph.D. Thesis, Ohio State University (1990)

Liu Xulin, "An expert system for casting defect analysis", *Special Casting and Nonferrous Alloys*, No.4, p.34-36 (2000)

Lok S.M., C.M. Wong and T.P. Leung, "Plastic mould parting directions determination with feature recognition", *Proceedings of the International Manufacturing Conference in China*, p.557-562, Harbin, China, 19-21 Oct. (1995)

Louis, L. and draper, A. B., "Effect of overflow well gating and injection parameter on the porosity of die castings", *Modern Casting*, June, pp35-38 (1966)

Lu H.Y. and Lee W.B., "A simplified approach for simulation of metal flow in cylindrical die casting", *Journal of Materials Processing Technology*, 91, pp116-120 (1999)

Lu Hong Yuan, "Computer simulation of metal flow I thin wall diecasting dies", Mphil Thesis, Hong Kong Polytechnic University (1993)

Lu H.Y. and Lee W.B., "A simplified approach for simulation of metal flow in cylindrical sleeve die casting cavity", *Journal of Materials Processing Technology*, 91, p.1116-1201 (1999)

Lu H.Y. and Lee W.B., "Detection of interferential elements and release directions in die-cast components", *Journal of Engineering Manufacture*, Part B, Vol 214, pp431-441 (2000a)

Lu H.Y. and Lee W.B., "An automatic approach to parting scheme design for die casting", *Proceedings of 2ed International die casting conference*, Shanghai, China, April 24-28, pp351-367 (2000b)

Lu H.Y. and Lee W.B., "Construction of DCES expert system for die casting defects and their countermeasure", *Foundry*, Vol.50, No.4, pp201-205 (2001)

Luo Peng and Zhou Jijin, "Fuzzy evaluation for quality insurance system of net shape manufacturing", *Foundry Technology*, No.1, p.32-33 (2001)

Hill J.L., S.K. Manna, A.K. Das and T.S. Piwonka, "Knowledge-based design of rigging systems for investment castings", *AFS Transactions*, 101, p.1097-1102 (1993)

Mitsushi Ohmasa and Itsuo Ohnaka, "Computer simulation of mold filling by direct finite difference method", *Foundry*, Vol.63, No.10, p.817-822 (1991)

NADCA, *Die Casting Technology*, North American Die Casting Association, Rosemont, Illinois 60018-4721, USA (1996)

Nagasaka Y., S. Kiguchi and Nachi, "Three-dimensional computer simulation of casting process", *AFS Transactions*, 89-117, p.553-563 (1989)

Nainy-Nejad M., R.C. Creese and B.Ravi, "Research issues in computer-aided parting design for casting", *AFS Transactions*, 97-86, p.545-550 (1997)

Nanda N.K., K.A. Smith, K.F. Haberle and V.R. Voller, "A knowledge-based computer tool for casting process design", *Journal of Metals*, 46(9) p.27-30, September (1994)

Ngoi K A, S W Lye and J Chen, "Analysising the natural resting aspect of a complex shaped part on a hard surface for automated parts feeding", *Proc Instn Mech Engrs*, Vol.211 Part B, p.453-458 (1997)

Nichols, B.D., Hirt, C.W. and Hotchkiss, R.S., "SOLA-VOF: A solution algorithm for transient fluid flows with multiple free boundaries", *Los Alamos Scientific Laboratory Report* LA-8355 (1980)

Nyamekye K.An, Bain. Cunningham and Askeland Ramsay, "Expert system for designing gating system for permanent mold tilt-pour casting process", *AFS Transactions*, 94-150, pp127-131 (1994)

Ohtsuka Y., T. Ono, "Computer simulation system of molten metal flow in die casting", *57th World Foundry Congress*, Osaka, 23-28, September, No.29, p.3-10 (1990)

Patankar Suhas V., *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corporatin-Washington (1980)

Pillinger I., P. Hartley, and T.A. Dean, "An intelligent knowledge-based system for the design of forging dies", *Artificial Intelligence in Design*, Editted by D.T. Pham, Springer-Verlag London Limited, p.319-341 (1991)

Plyatskii V.M., "Principles of the design of gating systems for pressure die casting", *Russian Casting Production*, Jan., p.12-15 (1961)

Rao P.N., R. Thukral and K.L. Rajpal, "Computer-aided design of gating systems for die casting dies" *15th Die Casting Congress & Exposition*, Paper No. G-T89_064, p.1-8, Oct. 16-19, St. Louis, MO (1989)

Ravi B and M N Srinivasan, "Decision criteria for computer-aided parting surface design" *Computer-Aided Design*, Vol.22, Number 1, Jan/Feb, p.11-18 (1990)

Roshan H. Md., K. Sudesh, "Expert system for analysis of casting defects: "cause" module", *AFS Transections*, 89-113, p.601-606 (1989)

Roshan H M, Jain P., "A knowledge representation for diagnosing defects in steel castings", *AFS Transections*, 90-101, p.477-482 (1990)

Shen Hong, *Mechanical Engineering Handbook*, Mechanical Industry Press, Beijing (1982)

Shi Tingzao, Wang Yuwei, *Foundry Handbook*, North-East University Press, Shenyang (1988)

Shivpuri R. and M. Kuthirakulathu, "Nonisothermal 3-D finite difference simulation of cavity filling during the die casting process", AMD-Vol. 125, *Advances in Finite Deformation Problems in Materials processing and Structures*, p.75-88, ASME (1991)

Yuji Hishinuma and Ho Siauw, "The application of computer-aided design tools for pressure diecasting", *Proceedings of the First China International Die Casting Congress*, 14-18, Beijing, China, p.326-331 (1997)

Sirilertworakul N., P.D. webster and T.A. Dean, "A software package for the design and production of castings", 60th World Foundry Congress, 26 September/1 October, Hague, p.14.1-14.10 (1993a)

Sirilertworakul N., P.D. webster and T.A. Dean, "A Knowledge base for alloy and process selection for casting", *International Journal of Machine Tools and Manufacture*, Vol. 33, No.3, p.401-416 (1993b)

Stoehr, R.A. and Hwang C., "Coupled heat transfer and fluid flow in the filling of castings", *AFS Transactions*, 88-176, p.733-740 (1988)

Sudesh K., G.R. Prakash and K.M. Roshan, "Knowledge-based expert system for analysis of casting defects", *AFS Transactions*, 96, p.145-150 (1988)

Sulaiman B. and T. Gethin, "A network technique for metal flow analysis in the filling system of pressure diecasting and its experimental verification on a cold chamber machine", *Journal of Engineering Manufacture*, Vol. 206 p.261-275 (1992)

Sulaiman Shamsuddin and Tham Chee Keen, "Flow analysis along the runner and gating system of a casting process", *Journal of Materials Processing Technlogy*, 63, p.690-695 (1997)

Sun Jiaguang & Yang Changgui, *Computer Graphics*, Qinghua University Press (1995)

Tai C.C, J.C Lin, "A runner-optimization design study of a diecasting die", *Journal of Materials Processing Technology*, 84, p.1-12 (1998)

Tai C.C, J.C Lin, "Optimal Position for the Injection Gate of a diecasting die", *Journal of Materials Processing Technology*, 86, p.87-100 (1999)

Tan S T, M F Yuen and W S Sze et al, "Parting lines and parting surfaces of injection moulded parts", *Proceedings of The Institution of Mechanical Engineers*, p.211-221, Vol.204 (1990)

Thukkaram, P., "Factors controlling design of runners, gates, overflows and vents in die casting dies", 7^{th} *SDCE International Die Casting Congress*, Chicago (1972)

Upadhya G. and A.J Paul, "Comprehensive casting analysis model using a geometry-based technique followed by fully coupled 3-D fluid flow, heat transfer and solidification kinetics calculations" *AFS Transactions*, 92-118, p.925-933 (1992)

Vellapillil Gourisankar, "Computer modeling of parting plane problem", M.Eng Thesis, Georgia Institute of Technology (1985)

Venkatesan K. and R. Shivpuri, "Numerical simulation of die cavity filling in die castings and an evaluation of process parameters on die wear", *Die Casting Technology*, NADCA, October 18-21, Cleveland T93-011 (1993)

Venus W., *ANSCHNITTEHNIK FUR DRUCKGUSS*, Giesserei-Verlag GmbH (1975)

Wang Caihua, Song Liantian, *Fuzzy Set Methodology*, Construction Industry Press, Beijing (1987)

Wang Qi, *Practicable Fuzzy Mathematics*, Science & Technology Press. Beijing (1992)

Wang Qun, Ming Rao and Ji Zhou, "An intelligent approach to conceptual design of mechanical system", *Artificial Intelligence in Optimal Design and Manufacturing*/Zuomin Dong, PTR Prentice Hall, New Jersey, p.59-82 (1994)

Wei T.J. (尉腾蛟), *第五代-日本第五代电脑对世界的冲击*，中国友谊出版社，北京（1985）

Weinbender Rex, "Virtual foundry answers 'What If' production questions", *Modern Casting*, (10) p.24-26 (1997)

Weinstein M., S. Manoochehri, "Geometric influence of a molded part on the draw direction range and [parting line location", *Journal of Mechanical Design*, March, Vol. 118, p.29-39 (1996)

Wilson Graham, *A Die Casting Calculating Tool-DC-CALC*, HotFlo Company (2001)

Woo, Tony C., "Visibility maps and spherical algorithms", *Computer-Aided Design*, 26(1), pp6-16 (1994)

Wu Wei, "Calculation of withdrawal force of core in die casting die", *Nonferrous Alloy and Special Casting*, No.6, p.32-33 (1987)

Wuerger D. Gadh R., "Virtual prototyping of die design part one: Theory and formulation", *Journal of Concurrent Engineering: Research and applications*, 5(4):307-15 (1997)

Xu Hong, Jin Yuchun, Cheng Jun and Tian Guanghui, "Numerical simulation of 3-D thermal field for die casting dies", *Proceedings of the First China International Die Casting Congress*, 14-17 April, 1997, Beijing, p.386-390 (1997)

Yang Shuxiang and Yao Tingxiu, "Calculation of ingate area in die design", *Nonferrous Alloy and Special Casting*, No.4, p.32-35 (1980)

Yin ZhouPing, Han Ding, YouLun Xiong, "Virtual prototyping of mould design: geometric mouldability analysis for near-net-shape manufactured parts by feature recognition and geometric reasoning", *Computer-Aided Design*, 33 (2001) 137-154 (2001)

Yoshiaki Yamamoto, Yasushi Iwata and Motoyuki Nakamira, "Molten metal velocities and gas pressure in die cavity and defects in commercial aluminum pressure die castings", *15th International Die Casting Congress& Exposition*, Paper No. G-T89-081, p.1-8, Oct. 16-19, St. Louis, Missouri (1989)

Yuan Haoyang, Min Guangguo and et al, "Study on practical 3-D geometric modeling method for solidification simulation", *Foundry*, No.2, pp8-11 (1994)

Yue Shuhua, Zhang Yifeng, Lu Keming, "CAD/CAE/CAM for Diecasting Mould", *Foundry*, No.1, p.15-18 (1997)

Yukio Ohtsuka, Takaoki Ono, Kuniaki Mizuno and Eikichi Matsubara, "Computer simulation system of molten metal flow in die casting", *15th Die Casting Congress & Exposition*, Paper No. G-T89-042, Oct. 16-19, 1989, St. Louis, MO (1989)

Zadeh, L.A., "Fuzzy sets as a basis for a theory of possibility", *Fuzzy Sets and System* 1, p.3-28 (1978)

Zhan Hongyuan, "Die casting technology (2)", *Special Casting and Nonferrous Alloy*, No.6, p.51-55 (1989)

Zhang K.F, A.J.Wright and B.J.Davies, "A feature-recognition knowledge base for process planning of rotational mechanical components", *The International Journal of Advanced Manufacturing Technology*, Vol. 4, No.1 p.13-25 (1989)

Zhang Weishan, Xiong Shoumei, Liu Baicheng, "3-D numerical simulation of temperature field for die casting", *Special Casting and Nonferrous Alloys*, No. 6, p.23-27 (1996)

Zhang Weishan, Xiong Shoumei, Liu Baicheng, "Study on a CAD/CAE system of die casting", *Journal of Materials Processing Technology*, 63, p.707-711 (1997)

Zheru Chi, Hong Yan, Tuan Pham, *FUZZY ALGORITHMS*, World Scientific (1996)

Appendices

Appendix 1 Algorithmic basis to solve for release direction

Release direction (RD) of a concave region (CR) is dependent on the RD of a number of relative points on the CR surfaces. The RD of a point corresponds to a hemispherical orientation on RD map. The hemispherical orientation, or the RD of a point can be determined with the normal vector at the point, and equals to the angular deviation of $\pi/2$ from the normal vector. The RD of a CR is the intersection of RDs of all relative points on the CR surfaces. Once the normal vector at each relative point is obtained, the intersection can be solved. An algorithm which can both solve the RD and record the RD intersection is required. Such an algorithm is developed based on the principle of the coordinate transformation. The algorithmic basis is discussed below.

Rotation transformation

Let $OX_1Y_1Z_1$ be the system coordinate obtained from the OXYZ system by a rotation transformation with a pair of angles (Ψ_1, θ_1), and the X_1, Y_1 and Z_1 can be expressed in the OXYZ system as:

X_1:
$$\begin{aligned} X &= X_1\cos\Psi_1\cos\theta_1 \\ Y &= X_1\sin\Psi_1\cos\theta_1 \\ Z &= X_1\sin\theta_1 \end{aligned} \tag{A1-1}$$

Y_1:
$$\begin{aligned} X &= Y_1\sin\Psi_1 \\ Y &= Y_1\cos\Psi_1 \\ Z &= 0 \end{aligned} \tag{A1-2}$$

Z_1: $$X = Z_1\cos\Psi_1\sin\theta_1$$
$$Y = Z_1\sin\Psi_1\sin\theta_1 \qquad \text{(A1-3)}$$
$$Z = Z_1\cos\theta_1$$

The system coordinate used in rotation transformation is shown in Fig. A1.1.

Secondary rotation transformation

Let $OX_2Y_2Z_2$ be the coordinate system obtained from the $OX_1Y_1Z_1$ by a rotation of (Ψ_2, θ_2). According to Equation (A1-3), Z_2 can be expressed in the $OX_1Y_1Z_1$ as:

Z_2: $$X_1 = Z_2\cos\Psi_2\sin\theta_2$$
$$Y_1 = Z_2\sin\Psi_2\sin\theta_2 \qquad \text{(A1-4)}$$
$$Z_1 = Z_2\cos\theta_2$$

The $OX_1Y_1Z_1$ is the system coordinate obtained from the OXYZ by a rotation of (Ψ_1, θ_1). As the same, X_1, Y_1 and Z_1 can be expressed in the OXYZ respectively as:

X_1: $$X = X_1\cos\Psi_1\cos\theta_1 = Z_2\cos\Psi_1\cos\theta_1\cos\Psi_2\sin\theta_2$$
$$Y = X_1\sin\Psi_1\cos\theta_1 = Z_2\sin\Psi_1\cos\theta_1\cos\Psi_2\sin\theta_2 \qquad \text{(A1-5)}$$
$$Z = X_1\sin\theta_1 = Z_2\sin\theta_1\cos\Psi_2\sin\theta_2$$

Y_1: $$X = Y_1\sin\Psi_1 = Z_2\sin\Psi_1\sin\Psi_2\sin\theta_2$$
$$Y = Y_1\cos\Psi_1 = Z_2\cos\Psi_1\sin\Psi_2\sin\theta_2 \qquad \text{(A1-6)}$$
$$Z = 0$$

Z_1: $$X = Z_1\cos\Psi_1\sin\theta_1 = Z_2\cos\Psi_1\sin\theta_1\cos\theta_2$$

$$Y = Z_1\sin\Psi_1\sin\theta_1 = Z_2\sin\Psi_1\sin\theta_1\cos\theta_2 \qquad \text{(A1-7)}$$

$$Z = Z_1\cos\theta_1 = Z_2\cos\theta_1\cos\theta_2$$

Adding Equations (A1-5), (A1-6) and (A1-7), Z_2 can be expressed in the OXYZ as:

Z_2: $$X = Z_2\,(\cos\Psi_1\cos\theta_1\cos\Psi_2\sin\theta_2 + \sin\Psi_1\sin\Psi_2\sin\theta_2 + \cos\Psi_1\sin\theta_1\cos\theta_2)$$

$$Y = Z_2\,(\sin\Psi_1\cos\theta_1\cos\Psi_2\sin\theta_2 + \cos\Psi_1\sin\Psi_2\sin\theta_2 + \sin\Psi_1\sin\theta_1\cos\theta_2) \qquad \text{(A1-8)}$$

$$Z = Z_2\,(\sin\theta_1\cos\Psi_2\sin\theta_2 + \cos\theta_1\cos\theta_2)$$

Equivalent rotation transformation

From Equation (A1-8), we have,

$$\theta = \arccos(Z) \qquad \text{(A1-9)}$$
$$\Psi = \mathrm{arctg}\,(Y/X)$$

Where (Ψ, θ) means that the coordinate transformation from the OXYZ to the $OX_2Y_2Z_2$ system by a rotation of (Ψ, θ) is equivalent to that from the OXYZ to $OX_1Y_1Z_1$ by a rotation of (Ψ_1, θ_1), then to the $OX_2Y_2Z_2$ from the $OX_1Y_1Z_1$ by a rotation of (Ψ_2, θ_2), i.e.,

$$OXYZ \longrightarrow (\Psi, \theta) \longrightarrow OX_2Y_2Z_2$$

is equivalent to

$$OXYZ \longrightarrow (\Psi_1, \theta_1) \longrightarrow OX_1Y_1Z_1 \longrightarrow (\Psi_2, \theta_2) \longrightarrow OX_2Y_2Z_2$$

Application of rotation transformation in solving for release direction

Suppose there are two normal vectors, Z_1' and Z_1'', at two relative points of CR surface in the system coordinate OXYZ. Let $OX_1'Y_1'Z_1'$ be the system coordinate obtained from the $OX_1Y_1Z_1$ by a rotation of (Ψ_1', θ_1'), i.e.,

$$OXYZ \longrightarrow (\Psi_1', \theta_1') \longrightarrow OX_1'Y_1'Z_1'$$

According to Equation (A1-3), Z_2 can be expressed in the $OX_1Y_1Z_1$ as:

$$Z_1': \quad \begin{aligned} X &= Z_1'\cos\Psi_1'\sin\theta_1' \\ Y &= Z_1'\sin\Psi_1'\sin\theta_1' \\ Z &= Z_1'\cos\theta_1' \end{aligned} \tag{A1-10}$$

Similarly, let $OX_1''Y_1''Z_1''$ be the system coordinate obtained from the OXYZ by a rotation of (Ψ_1'', θ_1''), i.e.,

$$OXYZ \longrightarrow (\Psi_1'', \theta_1'') \longrightarrow OX_1''Y_1''Z_1''$$

then

$$Z_1'': \quad \begin{aligned} X &= Z_1''\cos\Psi_1''\sin\theta_1'' \\ Y &= Z_1''\sin\Psi_1''\sin\theta_1'' \\ Z &= Z_1''\cos\theta_1'' \end{aligned} \tag{A1-11}$$

Let S_1' be a hemisphere orientation determined by Z_1' in the system coordinate $OX_1'Y_1'Z_1'$. Consider that $OX_2'Y_2'Z_2'$ is the system coordinate obtained from the $OX_1'Y_1'Z_1'$ by a rotation of (Ψ_2', θ_2'), i.e.,

$OX_1'Y_1'Z_1' \longrightarrow (\Psi_2', \theta_2') \longrightarrow OX_2'Y_2'Z_2'$

$$S_1' = \sum_{\psi_2'=0}^{360} \sum_{\theta_2'=0}^{90} OX_2'Y_2'Z_2'(\psi_2', \theta_2') \qquad \text{(A1-12)}$$

where $0° <= \Psi_2' <= 360°$, $0° <= \theta_2' <= 90°$ are in the $OX_1'Y_1'Z_1'$system.

Let S' and S_1'be the hemisphere orientation in the OXYZ, from Equations (A1-10), (A1-12) and (A1-8), we have,

$$S' = \sum_{\psi', \theta'} OX'Y'Z' \qquad \text{(A1-13)}$$

where

$\theta' = \arccos(Z')$

$\Psi' = \mathrm{arctg}\,(Y'/X')$

and

$X' = Z_2'\,(\cos\Psi_1'\cos\theta_1'\cos\Psi_2'\sin\theta_2' + \sin\Psi_1'\sin\Psi_2'\sin\theta_2' + \cos\Psi_1'\sin\theta_1'\cos\theta_2')$

$Y' = Z_2'\,(\sin\Psi_1'\cos\theta_1'\cos\Psi_2'\sin\theta_2' + \cos\Psi_1'\sin\Psi_2'\sin\theta_2' + \sin\Psi_1'\sin\theta_1'\cos\theta_2')$

$Z' = Z_2'\,(\sin\theta_1'\cos\Psi_2'\sin\theta_2' + \cos\theta_1'\cos\theta_2')$

with $0° <= \Psi_2' <= 360°$, $0° <= \theta_2' <= 90°$ in the $OX_1'Y_1'Z_1'$ system.

As the same as S', S'' can be expressed in OXYZ as,

$$S'' = \sum_{\psi'', \theta''} OX''Y''Z'' \qquad \text{(A1-14)}$$

where

$\theta'' = \arccos(Z'')$

$\Psi'' = \text{arctg}\,(Y''/X'')$

and

$X'' = Z_2''\,(\cos\Psi_1''\cos\theta_1''\cos\Psi_2''\sin\theta_2'' + \sin\Psi_1''\sin\Psi_2''\sin\theta_2'' + \cos\Psi_1''\sin\theta_1''\cos\theta_2'')$

$Y'' = Z_2''\,(\sin\Psi_1''\cos\theta_1''\cos\Psi_2''\sin\theta_2'' + \cos\Psi_1''\sin\Psi_2''\sin\theta_2'' + \sin\Psi_1''\sin\theta_1''\cos\theta_2'')$

$Z'' = Z_2''\,(\sin\theta_1''\cos\Psi_2''\sin\theta_2'' + \cos\theta_1''\cos\theta_2'')$

with $0° <= \Psi_2'' <= 360°$, $0° <= \theta_2'' <= 90°$ in the $OX_1''Y_1''Z_1''$system.

Compare with all (θ',Ψ') and (θ'',Ψ'') orientations, the intersection aggregation of S′ and S″ is available. Based on the rotation transformation, the algorithm which can both solve the RD and record the RD intersection is developed.

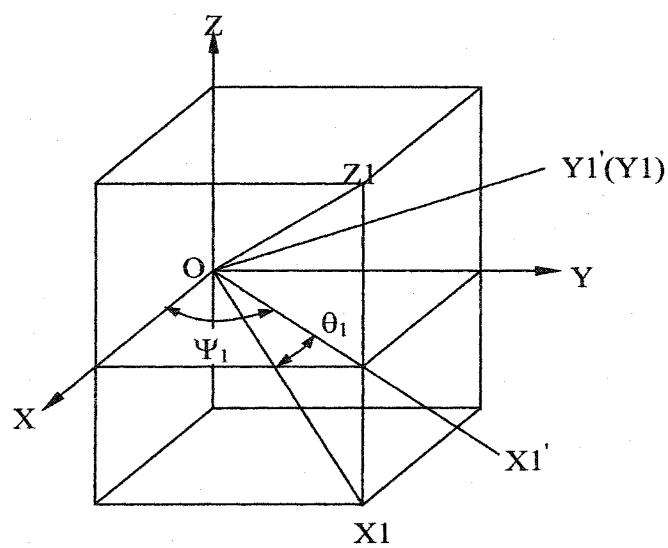


Fig. A1.1 The system coordinate used in coordinate rotation transformation

Appendix 2 Some examples in case library

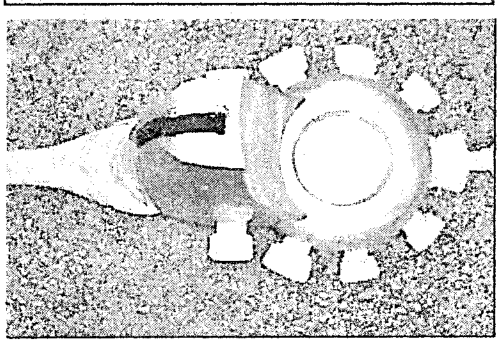
Case-based reasoning is based on the idea of making use of solutions to previous problems for solving new one. In the case-based gating design system, the knowledge is represented as different gating cases. With reference to the matched case, a new gating configuration can be designed. A case stored in the case library includes two basic segments, i.e., feature description and gating solution. In addition, a case index is required for retrieval of a specific case. Three typical case examples of aluminum, zinc and magnesium are shown in Fig. A2.1, A2.2 and A2.3 respectively.

Casing body

Indexing identifier
{parting feature: sleeve & sphere, circular}

Alloy: Al alloy
Casting size:

Length	420.0 mm
Width	130.0 mm
Height	220.0 mm
Wall thickness	3.5 mm



Gating type: fan gate
linear depth
Connection: overlap
Ingate size:

Thickness	1.0 mm
Width	110.0 mm

Fan runner:

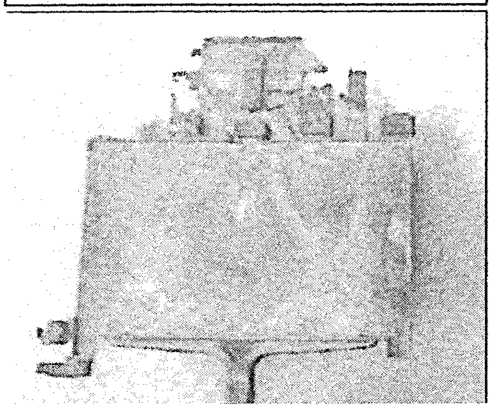
End	
Width	110.0 mm
Thickness	5.0 mm
Inlet	
Shape	trapezoid
Top width	36.0 mm
Bottom width	40.0 mm
Height	8.0 mm

Fig. A2.1 A casing body of aluminum alloy with a fan gate

Notebook computer cover

Indexing identifier
{parting feature: flat_plate, rectangular}

Alloy: Mg alloy
Casting size:
Length 250 mm
Width 200 mm
Thickness 1.5 mm



Gating type: tapered tangential runner
Connection: edge
Ingate size:
Length 210 mm
Thickness 0.8 mm
Gated runner:
End
Shape half trapezoid
Top width 3.2 mm
Bottom width 7.5 mm
Height 5.5 mm
Inlet
Shape trapezoid
Top width 7.0 mm
Bottom width 16.0 mm
Height 11.0 mm
Main runner:
Shape symmetric trapezoid
Top width 18 mm
Bottom width 24 mm
Height 14 mm
Vent:
Thickness 0.5 mm
Width 15.0 mm
Overflow:
Width 26.0 mm
Length 18.0 mm
Thickness 7.0mm

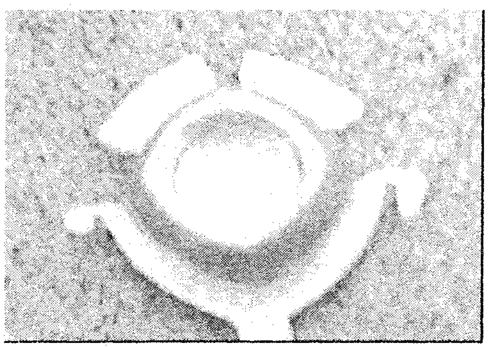
Fig. A2.2 A notebook computer cover of magnesium alloy with a tapered tangential runner

Container

Indexing identifier
{parting feature: sleeve, rectangular}

Alloy: Zn alloy
Casting size:

Length	60 mm
Width	60 mm
Height	40mm
Inner diameter	50mm



Gating type: tapered tangential runner
Connection: overlap
Ingate size:

Thickness	0.3-0.4 mm
Width	100.0 mm

Gated runner:

End	
Shape	half trapezoid
Top width	5.0 mm
Bottom width	7.0 mm
Height	4.0 mm
Inlet	
Shape	symmetric trapezoid
Top width	8.0 mm
Bottom width	12.0 mm
Height	5.0 mm

Main runner:

Shape	symmetric trapezoid
Top width	10 mm
Bottom width	23 mm
Height	5.5 mm

Fig. A2.3 A container casting of zinc alloy with a tapered tangential runner

Appendix 3 Program for construction of a die casting model

Program for construction of a die casting model includes the casting model generation, display and graph processing, and is written with Visual C++. In this section, a program list and a system interface are given.

```
// MainFrm.cpp : implementation of the CMainFrame class
#include "stdafx.h"
#include "Casting.h"
#include "Dialog1.h"
#include "MainFrm.h"
#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

IMPLEMENT_DYNCREATE(CMainFrame, CFrameWnd)
BEGIN_MESSAGE_MAP(CMainFrame, CFrameWnd)
	//{{AFX_MSG_MAP(CMainFrame)
	ON_WM_CREATE()
	ON_COMMAND(ID_PARAMETER, OnParameter)
	//}}AFX_MSG_MAP
	// Global help commands
	ON_COMMAND(ID_HELP_FINDER, CFrameWnd::OnHelpFinder)
	ON_COMMAND(ID_HELP, CFrameWnd::OnHelp)
	ON_COMMAND(ID_CONTEXT_HELP, CFrameWnd::OnContextHelp)
	ON_COMMAND(ID_DEFAULT_HELP, CFrameWnd::OnHelpFinder)
END_MESSAGE_MAP()


static UINT indicators[] =
{
	ID_SEPARATOR,		// status line indicator
	ID_INDICATOR_CAPS,
	ID_INDICATOR_NUM,
	ID_INDICATOR_SCRL,
};
//////////////////////////////////////////////////////////////////////////////
// CastingView.cpp : implementation of the CCastingView class
#include "stdafx.h"
#include "gl\glu.h"
#include "gl\glaux.h"
#include "math.h"
#include "Casting.h"
#include "Globals.h"
#include "CastingDoc.h"
#include "CastingView.h"
#include "VoxelStateDlg.h"
```

```
#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif
#define VIEW_NONE         -1
#define VIEW_ZOOM         0
#define VIEW_PAN          1
#define VIEW_ROTATE              2
#define VIEW_ZOOMTO              3
#define VIEW_ZOOMIN              4
#define VIEW_ZOOMOUT      5
#define VIEW_RESET        6

BOOL gbBusy;
/////////////////////////////////////////////////////////////////////////////
// CCastingView
IMPLEMENT_DYNCREATE(CCastingView, CView)
BEGIN_MESSAGE_MAP(CCastingView, CView)
        //{{AFX_MSG_MAP(CCastingView)
        ON_WM_CREATE()
        ON_WM_DESTROY()
        ON_WM_KEYDOWN()
        ON_WM_KEYUP()
        ON_WM_SIZE()
        ON_WM_LBUTTONDOWN()
        ON_WM_LBUTTONUP()
        ON_WM_MOUSEMOVE()
        ON_COMMAND(IDM_VIEW_NONE, OnViewNone)
        ON_UPDATE_COMMAND_UI(IDM_VIEW_NONE, OnUpdateViewNone)
        ON_COMMAND(IDM_VIEW_PAN, OnViewPan)
        ON_UPDATE_COMMAND_UI(IDM_VIEW_PAN, OnUpdateViewPan)
        ON_COMMAND(IDM_VIEW_RESET, OnViewReset)
        ON_UPDATE_COMMAND_UI(IDM_VIEW_RESET, OnUpdateViewReset)
        ON_COMMAND(IDM_VIEW_ROTATE, OnViewRotate)
        ON_UPDATE_COMMAND_UI(IDM_VIEW_ROTATE, OnUpdateViewRotate)
        ON_COMMAND(IDM_VIEW_ZOOM, OnViewZoom)
        ON_UPDATE_COMMAND_UI(IDM_VIEW_ZOOM, OnUpdateViewZoom)
        ON_COMMAND(IDM_VIEW_ZOOMIN, OnViewZoomin)
        ON_UPDATE_COMMAND_UI(IDM_VIEW_ZOOMIN, OnUpdateViewZoomin)
        ON_COMMAND(IDM_VIEW_ZOOMOUT, OnViewZoomout)
        ON_UPDATE_COMMAND_UI(IDM_VIEW_ZOOMOUT, OnUpdateViewZoomout)
        ON_COMMAND(IDM_VIEW_ZOOMTO, OnViewZoomto)
        ON_UPDATE_COMMAND_UI(IDM_VIEW_ZOOMTO, OnUpdateViewZoomto)
        ON_COMMAND(ID_LIGHT_LEFT, OnLightLeft)
        ON_COMMAND(ID_LIGHT_RIGHT, OnLightRight)
        ON_COMMAND(ID_LIGHT_RAISE, OnLightRaise)
        ON_COMMAND(ID_LIGHT_LOWER, OnLightLower)
        ON_COMMAND(IDM_PART_PARTMODEL, OnPartPartModel)
        ON_COMMAND(IDM_PART_VOXEL, OnPartVoxel)
        ON_COMMAND(IDM_PART_DPART, OnPartDPart)
        ON_COMMAND(IDM_PART_DVOXEL, OnPartDVoxel)
```

```
	ON_COMMAND(IDM_PART_READ, OnPartRead)
	ON_COMMAND(IDM_PART_SAVE, OnPartSave)
	ON_COMMAND(IDM_MOULD_PART, OnMouldPart)
	ON_COMMAND(IDM_MOULD_INSERT, OnMouldInsert)
	ON_COMMAND(IDM_TOOLS_SETSTATE, OnToolsSetState)
	//}}AFX_MSG_MAP
	// Standard printing commands
	ON_COMMAND(ID_FILE_PRINT, CView::OnFilePrint)
	ON_COMMAND(ID_FILE_PRINT_DIRECT, CView::OnFilePrint)
	ON_COMMAND(ID_FILE_PRINT_PREVIEW, CView::OnFilePrintPreview)
END_MESSAGE_MAP()

/////////////////////////////////////////////////////////////////////////////
// CCastingView construction/destruction
CCastingView::CCastingView()
{
	// accelerating flag for view transformation and Light Orientation
	m_bShiftPressed = FALSE;
	/////////////////////////////////////////////
	// view transformation and default range of view window
	m_FirstPick = FALSE;
	m_XOrg = -1200;			m_YOrg = -1200;
	m_ZMax = 10000;
	m_XMax = 1200;			m_YMax = 1200;
	m_RX  = 0.0f; //30.0f;
	m_RY = 0.0f; //-30.0f;
	m_CurViewFunc = VIEW_NONE;
	// view transformation
	/////////////////////////////////////////////
	// Light Orientation {
	m_Light0Pos[0] = 1.0f;
	m_Light0Pos[1] = 1.0f;
	m_Light0Pos[2] = 1.0f;
	m_Light0Pos[3] = 0.0f;
	m_Light0Azimuth = 0.0f;
	m_Ligth0Elevation = 0.0f;
	// } Light Orientation
	m_Mould = NULL;
}
CCastingView::~CCastingView()
{
}
void CCastingView::OnDestroy()
{
	// restore the former environment
	if(wglGetCurrentContext() != NULL)
		wglMakeCurrent(NULL,NULL);
	if(m_hRC != NULL)
	{
		wglDeleteContext(m_hRC);
		m_hRC = NULL;
	}
```

```
        if(m_Mould)
                delete m_Mould;
        m_Mould = NULL;
        CView::OnDestroy();
}
BOOL CCastingView::PreCreateWindow(CREATESTRUCT& cs)
{
        cs.style |= WS_CLIPCHILDREN | WS_CLIPSIBLINGS;
        return CView::PreCreateWindow(cs);
}

#include "ViewFunc.h"
/////////////////////////////////////////////////////////////////////////////
// CCastingView drawing
void CCastingView::OnDraw(CDC* pDC)
{
        CCastingDoc* pDoc = GetDocument();
        ASSERT_VALID(pDoc);
        // Realize the PALETTE for the virtual environment
        // (Only once)
        if (m_hPalette)
        {
          SelectPalette(pDC->m_hDC, m_hPalette, FALSE);
          RealizePalette(pDC->m_hDC);
        }
        // Enter virtual environment
        // {
  wglMakeCurrent(pDC->m_hDC, m_hRC);
        // Clear display buffer
        glClearColor(1.0f, 1.0f, 1.0f, 0.0f);
  glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
        // Display scene begin {
        glPushMatrix();
                glRotated(m_RX, 1.0, 0.0, 0.0);
                glRotated(m_RY, 0.0, 1.0, 0.0);
                DrawWithOpenGL();
        glPopMatrix();
        ViewDraw();
        // } Dsiplay scene end
        // Realize display and so for animation
        SwapBuffers(pDC->m_hDC);
  wglMakeCurrent(pDC->m_hDC, NULL);
        // }
        // Exit virtual environment
}
void CCastingView::DrawWithOpenGL()
{
        if(gbBusy)
                return;
        gbBusy = TRUE;
        //////////////////////////////////////
        // Draw
```

```
        m_Mould->Display();
        // Draw
        //////////////////////////////////////
        glFlush();
        gbBusy = FALSE;
}
/////////////////////////////////////////////////////////////////////////////
// CCastingView printing
BOOL CCastingView::OnPreparePrinting(CPrintInfo* pInfo)
{
        // default preparation
        return DoPreparePrinting(pInfo);
}
void CCastingView::OnBeginPrinting(CDC* /*pDC*/, CPrintInfo* /*pInfo*/)
{
        // TODO: add extra initialization before printing
}
void CCastingView::OnEndPrinting(CDC* /*pDC*/, CPrintInfo* /*pInfo*/)
{
        // TODO: add cleanup after printing
}
/////////////////////////////////////////////////////////////////////////////
// CCastingView diagnostics
#ifdef _DEBUG
void CCastingView::AssertValid() const
{
        CView::AssertValid();
}
void CCastingView::Dump(CDumpContext& dc) const
{
        CView::Dump(dc);
}

CCastingDoc* CCastingView::GetDocument() // non-debug version is inline
{
        ASSERT(m_pDocument->IsKindOf(RUNTIME_CLASS(CCastingDoc)));
        return (CCastingDoc*)m_pDocument;
}
#endif //_DEBUG
/////////////////////////////////////////////////////////////////////////////
// CCastingView message handlers
int CCastingView::OnCreate(LPCREATESTRUCT lpCreateStruct)
{
        if (CView::OnCreate(lpCreateStruct) == -1)
                return -1;
        // Get directory Begin {
        char buffer[255];
        GetCurrentDirectory(255, buffer);
        gsCurrentDirectory = buffer;
        DWORD dwBuild;
        DWORD dwVersion = GetVersion();
        // Get major and minor version numbers of Windows
```

```
    DWORD dwWindowsMajorVersion =  (DWORD)(LOBYTE(LOWORD(dwVersion)));
    DWORD dwWindowsMinorVersion =  (DWORD)(HIBYTE(LOWORD(dwVersion)));
    // Get build numbers for Windows NT or Win32s
    if (dwVersion < 0x80000000)           // Windows NT
    {
            char str[MAX_PATH];
      if(dwWindowsMajorVersion == 5)  // Windows 2000
            {
                    dwBuild = (DWORD)(HIWORD(dwVersion));
                    GetWindowsDirectory(str, MAX_PATH);
                    gsTempDirectory = str;
                    gsTempDirectory += "\\temp";
            }
            else
            {
                    dwBuild = (DWORD)(HIWORD(dwVersion));
                    GetWindowsDirectory(str, MAX_PATH);
                    gsTempDirectory = str[0];
                    gsTempDirectory += ":\\temp";
            }
    }
    else if (dwWindowsMajorVersion < 4)       // Win32s
            dwBuild = (DWORD)(HIWORD(dwVersion) & ~0x8000);
    else      // Windows 95 -- No build numbers provided
    {
            dwBuild =  0;
            char str[MAX_PATH];
            GetWindowsDirectory(str, MAX_PATH);
            gsTempDirectory = str;
            gsTempDirectory += "\\temp";
    }
    // } Get directory End
    // initializate the pixel format
PIXELFORMATDESCRIPTOR pfd =
{
  sizeof(PIXELFORMATDESCRIPTOR), // Structure size.
  1,                  // Structure version number.
  PFD_DRAW_TO_WINDOW |        // Property flags.
    PFD_SUPPORT_OPENGL |
                      PFD_DOUBLEBUFFER,
  PFD_TYPE_RGBA,
  24,                  // 24-bit color.
  0, 0, 0, 0, 0, 0,        // Not concerned with these.
  0, 0, 0, 0, 0, 0, 0,      // No alpha or accum buffer.
  24,                  // 24-bit depth buffer.
  0, 0,                // No stencil or aux buffer.
  PFD_MAIN_PLANE,           // Main layer type.
  0,                  // Reserved.
  0, 0, 0              // Unsupported.
};
CClientDC clientDC(this);
int pixelFormat =
```

```
    ChoosePixelFormat(clientDC.m_hDC, &pfd);
  BOOL success =
    SetPixelFormat(clientDC.m_hDC, pixelFormat, &pfd);
  DescribePixelFormat(clientDC.m_hDC, pixelFormat,
    sizeof(pfd), &pfd);
  if (pfd.dwFlags & PFD_NEED_PALETTE)
    SetupLogicalPalette();
  m_hRC = wglCreateContext(clientDC.m_hDC);
        m_Mould = new CMould;
        ASSERT(m_Mould);
        return 0;
}
void CCastingView::SetupLogicalPalette()
{
  struct
  {
    WORD Version;
    WORD NumberOfEntries;
    PALETTEENTRY aEntries[256];
  } logicalPalette = { 0x300, 256 };
        BYTE reds[] = {0, 36, 72, 109, 145, 182, 218, 255};
        BYTE greens[] = {0, 36, 72, 109, 145, 182, 218, 255};
        BYTE blues[] = {0, 85, 170, 255};
  for (int colorNum=0; colorNum<256; ++colorNum)
  {
    logicalPalette.aEntries[colorNum].peRed =
      reds[colorNum & 0x07];
    logicalPalette.aEntries[colorNum].peGreen =
      greens[(colorNum >> 0x03) & 0x07];
    logicalPalette.aEntries[colorNum].peBlue =
      blues[(colorNum >> 0x06) & 0x03];
    logicalPalette.aEntries[colorNum].peFlags = 0;
  }
  m_hPalette = CreatePalette ((LOGPALETTE*)&logicalPalette);
}
/**************************************************************
 * SetupOrtho -- Set the viewport size of virtual environment *
 * ------------------------------------------------------ *
 * Parameter(s): void                                                    *
 * Comment: Default    size of system                                *
**************************************************************/
void CCastingView::SetupOrtho()
{
        CRect rect;
        GetClientRect(&rect);
        SetupOrtho(rect.Width(), rect.Height());

        Invalidate(FALSE);
}
/**************************************************************
 * SetupOrtho -- Set the viewport size of virtual environment *
 * Parameter(s):                                                         *
```

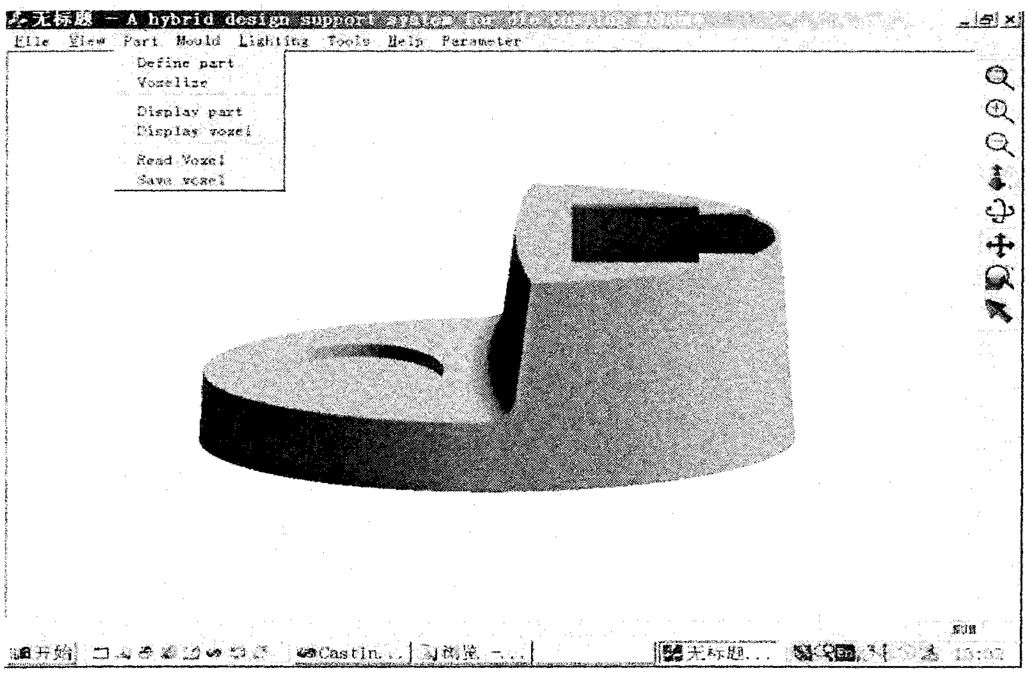
```
*   int cx: width of the new viewport                                    *
*   int cy: height of the new viewport                                   *
* -------------------------------------------------------- *
* Comment: Set viewport by users                                          *
*************************************************************/
void CCastingView::SetupOrtho(int cx, int cy)
{
        if( cx <= 0  || cy <= 0)
                return;
        CClientDC clientDC(this);
        wglMakeCurrent(clientDC.m_hDC,m_hRC);
        glViewport(0, 0, cx, cy);
        glMatrixMode(GL_PROJECTION);
        glLoadIdentity();
        GLfloat ratio;
        if(cx<=cy)
        {
                ratio = (GLfloat)cy/(GLfloat)cx;
                glOrtho(m_XOrg, m_XMax, m_YOrg*ratio, m_YMax*ratio, -10000.0f, 10000.0f);
        }
        else
        {
                ratio = (GLfloat)cx/(GLfloat)cy;
                glOrtho(m_XOrg*ratio, m_XMax*ratio, m_YOrg, m_YMax, -10000.0f, 10000.0f);
        }
        glMatrixMode(GL_MODELVIEW);
        glLoadIdentity();
        GLfloat light0Ambient[] = {0.3f, 0.3f, 0.3f, 1.0f};
        GLfloat light0Diffuse[] = {1.0f, 1.0f, 1.0f, 1.0f};
  //glLightModelfv(GL_LIGHT_MODEL_AMBIENT, light0Ambient);
        glLightfv(GL_LIGHT0, GL_AMBIENT, light0Ambient);
  glLightfv(GL_LIGHT0, GL_DIFFUSE, light0Diffuse);
        glLightfv(GL_LIGHT0, GL_POSITION, m_Light0Pos);
        glEnable(GL_DEPTH_TEST);
        glEnable(GL_COLOR_MATERIAL);
        glColorMaterial(GL_FRONT, GL_AMBIENT_AND_DIFFUSE);
  glEnable(GL_LIGHTING);
  glEnable(GL_LIGHT0);
        glTexEnvi(GL_TEXTURE_ENV,GL_TEXTURE_ENV_MODE,GL_DECAL);
        glTexParameteri(GL_TEXTURE_2D,GL_TEXTURE_WRAP_S, GL_REPEAT);
        glTexParameteri(GL_TEXTURE_2D,GL_TEXTURE_WRAP_T, GL_REPEAT);
        glTexParameteri(GL_TEXTURE_2D,GL_TEXTURE_MAG_FILTER,GL_LINEAR);
        glTexParameteri(GL_TEXTURE_2D,GL_TEXTURE_MIN_FILTER,GL_LINEAR);
/*      glEnable(GL_BLEND);
        glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA);
        glEnable(GL_LINE_SMOOTH);
        glEnable(GL_POINT_SMOOTH);
        glEnable(GL_POLYGON_SMOOTH);  */
        wglMakeCurrent(clientDC.m_hDC,NULL);
}
/*****************************************************************
* OnSize -- Process the windows message when its size is changed. *
```

```
*                          Called by system automatically.              *
* Parameter(s):
          *
*   int cx: width of the new viewport                                   *
*   int cy: height of the new viewport                                  *
* Comment: Standard MFC function                                        *
****************************************************************/
void CCastingView::OnSize(UINT nType, int cx, int cy)
{
	CView::OnSize(nType, cx, cy);

	SetupOrtho(cx, cy);
}
/*********************************************************************
* OnKeyDown -- Process the windows message when any key is pressed down. *
*                       Called by system automatically.             *
* Parameter(s): void
                 *
* Comment: Standard MFC function                                        *
*********************************************************************/
void CCastingView::OnKeyDown(UINT nChar, UINT nRepCnt, UINT nFlags)
{
	switch(nChar)
	{
	case VK_SHIFT:
		m_bShiftPressed = TRUE;			break;
	}

	CView::OnKeyDown(nChar, nRepCnt, nFlags);
}
/******************************************************************
* OnKeyUp -- Process the windows message when any key is released up. *
*                       Called by system automatically.            *
* Parameter(s): void
               *
* Comment: Standard MFC function                                       *
*******************************************************************/
void CCastingView::OnKeyUp(UINT nChar, UINT nRepCnt, UINT nFlags)
{
	switch(nChar)
	{
	case VK_SHIFT:
		m_bShiftPressed = FALSE;			break;
	}

	CView::OnKeyUp(nChar, nRepCnt, nFlags);
}
void CCastingView::OnPartPartModel()
{
	CString partmodel = gsCurrentDirectory + "\\Projects\\Table.EST"; //CutterHolder.EST";
	m_Mould->Def(partmodel);
	float minx = m_Mould->m_STLModel->m_Min[0];
```

```
        float miny = m_Mould->m_STLModel->m_Min[1];
        float maxx = m_Mould->m_STLModel->m_Max[0];
        float maxy = m_Mould->m_STLModel->m_Max[1];
        float width = maxx - minx;
        float height = maxy - miny;
        if(width > height)
        {
                m_XOrg = (float)(minx * 1.1);
                m_XMax = (float)(maxx * 1.1);
                m_YMax = (float)((maxy + (width-height)*0.5) * 1.1);
                m_YOrg = (float)((miny - (width-height)*0.5) * 1.1);
        }
        else
        {
                m_YOrg = (float)(minx * 1.1);
                m_YMax = (float)(maxx * 1.1);
                m_XMax = (float)((maxx + (height-width)*0.5) * 1.1);
                m_XOrg = (float)((minx - (height-width)*0.5) * 1.1);
        }
        SetupOrtho();
        OnPartDPart();
}
void CCastingView::OnPartVoxel()
{
        m_Mould->DefineVoxel();
        OnPartDVoxel();
}
void CCastingView::OnPartDPart()
{
        m_Mould->SetDisplayCase(11);
        Invalidate(FALSE);
}
void CCastingView::OnPartDVoxel()
{
        m_Mould->SetDisplayCase(1);
        Invalidate(FALSE);
}
void CCastingView::OnPartRead()
{
        if(m_Mould->ReadVoxelFromFile()== true)
        {
                OnMouldPart();
        }
}
void CCastingView::OnPartSave()
{
        m_Mould->WriteVoxelToFile();
}
void CCastingView::OnMouldPart()
{
        m_Mould->SetDisplayCase(1);
        Invalidate(FALSE);
```

```
}
void CCastingView::OnMouldInsert()
{
        m_Mould->SetDisplayCase(0);
        Invalidate(FALSE);
}
void CCastingView::OnToolsSetState()
{
        CVoxelStateDlg dlg;
        if(dlg.DoModal())
        {
                int i, j, k, state;
                dlg.GetValue(&i, &j, &k, &state);
                m_Mould->SetVoxelState(i, j, k, state);
                Invalidate(FALSE);
        }
}
```

Fig, A3. 1 shows an interface for construction of a die casting model.



Fig, A3. 1 An interface for construction of a die casting model

Appendix 4 Program for calculation of filling parameters

A fuzzy synthetic evaluation is used to calculate the filling parameters. The program list includes the calculation of fuzzy set, level weighting set, and evaluation set. Fig. A4.1 and Fig. A4.2 show two interfaces occurring during the calculation.

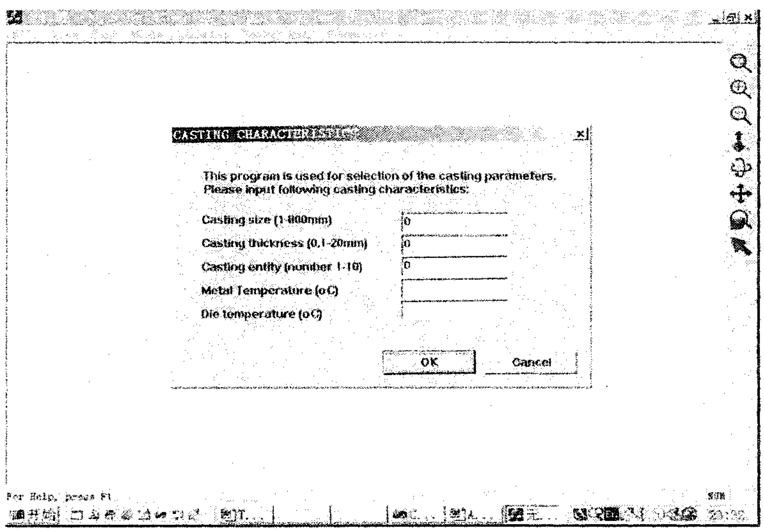
```
#include "stdafx.h"
#include "Casting.h"
#include "Dialog1.h"
#include "Membership.h"
#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif
/////////////////////////////////////////////////////////////////////////////
// CDialog1 dialog
CDialog1::CDialog1(CWnd* pParent /*=NULL*/)
	: CDialog(CDialog1::IDD, pParent)
{
	//{{AFX_DATA_INIT(CDialog1)
	m_castsize = 0.0f;
	m_wallsize = 0.0f;
	m_entitynumber = 0.0f;
	//}}AFX_DATA_INIT
}
void CDialog1::DoDataExchange(CDataExchange* pDX)
{
	CDialog::DoDataExchange(pDX);
	//{{AFX_DATA_MAP(CDialog1)
	DDX_Text(pDX, IDC_CASTSIZE, m_castsize);
	DDV_MinMaxFloat(pDX, m_castsize, 1.f, 800.f);
	DDX_Text(pDX, IDC_EDIT2, m_wallsize);
	DDV_MinMaxFloat(pDX, m_wallsize, 0.1f, 50.f);
	DDX_Text(pDX, IDC_EDIT3, m_entitynumber);
	DDV_MinMaxFloat(pDX, m_entitynumber, 1.f, 10.f);
	//}}AFX_DATA_MAP
}
BEGIN_MESSAGE_MAP(CDialog1, CDialog)
	//{{AFX_MSG_MAP(CDialog1)
	//}}AFX_MSG_MAP
END_MESSAGE_MAP()
/////////////////////////////////////////////////////////////////////////////
// CDialog1 message handlers
void CDialog1::OnOK()
{
	// TODO: Add extra validation here
//	CDialog::OnOK();
	UpdateData(true);
```

```
/*if (m_castsize<1||m_castsize>800)
{
        UpdateData(true);
        AfxMessageBox("!");
  CDialog1 dlg1;
        dlg1.DoModal();
}*/
// casting size section
        if (m_castsize>=1 && m_castsize<=100)
                SizeMembership[0]=1;
        else if (m_castsize >100 && m_castsize<500)
          SizeMembership[0]=((500-m_castsize)/400)*((500-m_castsize)/400)*((500-
m_castsize)/400);
        else if (m_castsize>=500)
          SizeMembership[0]=0;
  if (m_castsize>=1 && m_castsize<=190)
                SizeMembership[1]=(m_castsize/190)*(m_castsize/190)*(m_castsize/190);
        else if (m_castsize >190 && m_castsize<500)
          SizeMembership[1]=((500-m_castsize)/310)*((500-m_castsize)/310)*((500-
m_castsize)/310);
        else if (m_castsize>=500)
          SizeMembership[1]=0;

        if (m_castsize>=1 && m_castsize<=280)
                SizeMembership[2]=(m_castsize/280)*(m_castsize/280)*(m_castsize/280);
        else if (m_castsize >280 && m_castsize<500)
          SizeMembership[2]=((500-m_castsize)/220)*((500-m_castsize)/220)*((500-
m_castsize)/220);
        else if (m_castsize>=500)
          SizeMembership[2]=0;

        if (m_castsize>=1 && m_castsize<=370)
                SizeMembership[3]=(m_castsize/370)*(m_castsize/370)*(m_castsize/370);
        else if (m_castsize >370 && m_castsize<500)
          SizeMembership[3]=((500-m_castsize)/130)*((500-m_castsize)/130)*((500-
m_castsize)/130);
        else if (m_castsize>=500)
          SizeMembership[3]=0;

        if (m_castsize>=1 && m_castsize<460)
                SizeMembership[4]=(m_castsize/460)*(m_castsize/460)*(m_castsize/460);
        else if (m_castsize >=460)
          SizeMembership[4]=1;

//wall thickness section
        if (m_wallsize>=0.1 && m_wallsize<=0.8)
                WallMembership[0]=1;
        else if (m_wallsize >0.8 && m_wallsize<9.0)
          WallMembership[0]=((9.0-m_wallsize)/8.2)*((9.0-m_wallsize)/8.2)*((9.0-m_wallsize)/8.2);
        else if (m_wallsize>=9.0)
          WallMembership[0]=0;
```

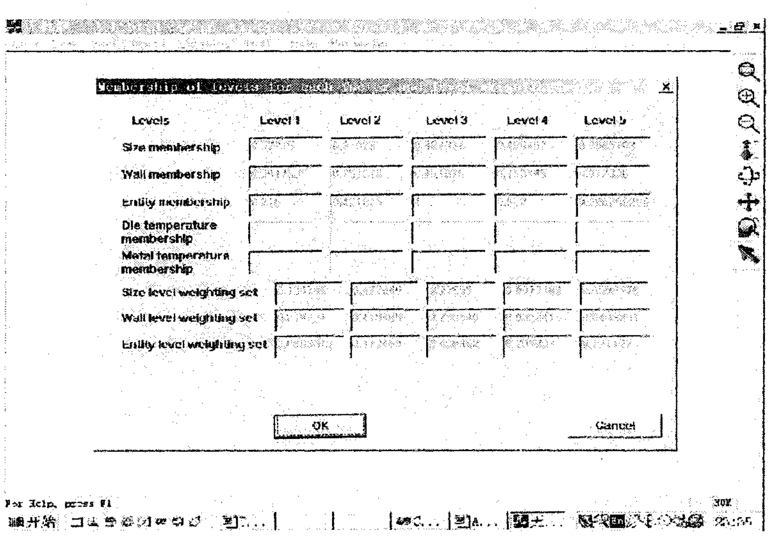
```
  if (m_wallsize>=0.1 && m_wallsize<=2.4)
              WallMembership[1]=(m_wallsize/2.4)*(m_wallsize/2.4)*(m_wallsize/2.4);
        else if (m_wallsize >2.4 && m_wallsize<9.0)
          WallMembership[1]=((9.0-m_wallsize)/6.6)*((9.0-m_wallsize)/6.6)*((9.0-m_wallsize)/6.6);
        else if (m_wallsize>=9.0)
          WallMembership[1]=0;

        if (m_wallsize>=0.1 && m_wallsize<=4.0)
              WallMembership[2]=(m_wallsize/4.0)*(m_wallsize/4.0)*(m_wallsize/4.0);
        else if (m_wallsize >4.0 && m_wallsize<9.0)
          WallMembership[2]=((9.0-m_wallsize)/5.0)*((9.0-m_wallsize)/5.0)*((9.0-m_wallsize)/5.0);
        else if (m_wallsize>=9.0)
          WallMembership[2]=0;

        if (m_wallsize>=0.1 && m_wallsize<=5.6)
              WallMembership[3]=(m_wallsize/5.6)*(m_wallsize/5.6)*(m_wallsize/5.6);
        else if (m_wallsize >5.6 && m_wallsize<9.0)
          WallMembership[3]=((9.0-m_wallsize)/2.4)*((9.0-m_wallsize)/2.4)*((9.0-m_wallsize)/2.4);
        else if (m_wallsize>=9.0)
          WallMembership[3]=0;

        if (m_wallsize>=0.1 && m_wallsize<=7.2)
              WallMembership[4]=(m_wallsize/7.2)*(m_wallsize/7.2)*(m_wallsize/7.2);
        else if (m_wallsize>=7.2)
          WallMembership[4]=1;

//entity section
  if (m_entitynumber>=1 && m_entitynumber<=2)
              EntityMembership[0]=1;
        else if (m_entitynumber>2 && m_entitynumber<7)
          EntityMembership[0]=((7-m_entitynumber)/5)*((7-m_entitynumber)/5)*((7-
m_entitynumber)/5);
        else if (m_entitynumber>=7)
          EntityMembership[0]=0;

        if (m_entitynumber>=1 && m_entitynumber<=3)
              EntityMembership[1]=(m_entitynumber/3)*(m_entitynumber/3)*(m_entitynumber/3);
        else if (m_entitynumber >3 && m_entitynumber<7)
          EntityMembership[1]=((7-m_entitynumber)/4)*((7-m_entitynumber)/4)*((7-
m_entitynumber)/4);
        else if (m_entitynumber>=7)
          EntityMembership[1]=0;

        if (m_entitynumber>=1 && m_entitynumber<=4)
              EntityMembership[2]=(m_entitynumber/4)*(m_entitynumber/4)*(m_entitynumber/4);
        else if (m_entitynumber >4 && m_entitynumber<7)
          EntityMembership[2]=((7-m_entitynumber)/3)*((7-m_entitynumber)/3)*((7-
m_entitynumber)/3);
        else if (m_entitynumber>=7)
          EntityMembership[2]=0;

  if (m_entitynumber>=1 && m_entitynumber<=5)
```

```
                EntityMembership[3]=(m_entitynumber/5)*(m_entitynumber/5)*(m_entitynumber/5);
        else if (m_entitynumber >4 && m_entitynumber<7)
          EntityMembership[3]=((7-m_entitynumber)/2)*((7-m_entitynumber)/2)*((7-
m_entitynumber)/2);
        else if (m_entitynumber>=7)
          EntityMembership[3]=0;

   if (m_entitynumber>=1 && m_entitynumber<6)
                EntityMembership[4]=(m_entitynumber/6)*(m_entitynumber/6)*(m_entitynumber/6);
        else if (m_entitynumber>=6)
          EntityMembership[4]=1;

// calculate weighting set

        float
sweight=(SizeMembership[0]+SizeMembership[1]+SizeMembership[2]+SizeMembership[3]+SizeMe
mbership[4]);
        float
wweight=(WallMembership[0]+WallMembership[1]+WallMembership[2]+WallMembership[3]+Wall
Membership[4]);
        float
eweight=(EntityMembership[0]+EntityMembership[1]+EntityMembership[2]+EntityMembership[3]+E
ntityMembership[4]);
        for (int i=0; i<5; i++)
        {
                Asizeweight[i]=SizeMembership[i]/sweight;
          Awallweight[i]=WallMembership[i]/wweight;
                Aentityweight[i]=EntityMembership[i]/eweight;
        }
        CMembership dlg;
        dlg.m_sizemembership1=SizeMembership[0];
   dlg.m_sizemembership2=SizeMembership[1];
        dlg.m_sizemembership3=SizeMembership[2];
        dlg.m_sizemembership4=SizeMembership[3];
        dlg.m_sizemembership5=SizeMembership[4];
        dlg.m_wallmembership1=WallMembership[0];
   dlg.m_wallmembership2=WallMembership[1];
        dlg.m_wallmembership3=WallMembership[2];
        dlg.m_wallmembership4=WallMembership[3];
   dlg.m_wallmembership5=WallMembership[4];
        dlg.m_entitymembership1=EntityMembership[0];
   dlg.m_entitymembership2=EntityMembership[1];
   dlg.m_entitymembership3=EntityMembership[2];
   dlg.m_entitymembership4=EntityMembership[3];
   dlg.m_entitymembership5=EntityMembership[4];
        dlg.m_sweight1=Asizeweight[0];
        dlg.m_sweight2=Asizeweight[1];
        dlg.m_sweight3=Asizeweight[2];
        dlg.m_sweight4=Asizeweight[3];
        dlg.m_sweight5=Asizeweight[4];
        dlg.m_eweight1=Aentityweight[0];
        dlg.m_eweight2=Aentityweight[1];
```

```
        dlg.m_eweight3=Aentityweight[2];
        dlg.m_eweight4=Aentityweight[3];
        dlg.m_eweight5=Aentityweight[4];
  dlg.m_wweight1=Awallweight[0];
        dlg.m_wweight2=Awallweight[1];
        dlg.m_wweight3=Awallweight[2];
        dlg.m_wweight4=Awallweight[3];
        dlg.m_wweight5=Awallweight[4];

        dlg.DoModal();
        CDialog::OnOK();
}
```



A4.1 An interface for casting characteristic input



A4.2 An interface of calculation display