

〈論 文〉

플라스틱 사출금형의 간섭 검사에 의한 사이드 코어의 설계

신기훈* · 이건우**

(1991년 11월 21일 접수)

Design of Side Cores of Plastic Injection Mold with Interference Check

Ki Hoon Shin and Kunwoo Lee

Key Words : Injection Mold(사출금형), Interference Check(간섭검사), Side Core(사이드코어)

Abstract

Eliminating the under-cut caused by interference between a mold and a product in designing a mold for injection molding processes is a very important problem. In general, the under-cut problem can be avoided by side cores which are the principal members of a mold assembly. In this research, a procedure has been developed by which the side cores and the corresponding core and cavity plates of a mold are generated after identifying the mold faces preventing product faces from moving while being discharged. The characteristic features of the procedure suggested in this paper are as follows. One is that the interference faces between the product and the mold are derived only from the core plate(or cavity plate) alone without considering the product together. The other is that the algorithm in the designing of side cores and modifying molds, is very efficient because it uses Euler operations instead of Boolean operations.

1. 서 론

일상 생활은 물론 산업 전반에 걸쳐 플라스틱 제품이 널리 사용됨에 따라 플라스틱 수지를 재료로 하는 여러가지 제품 가공 방법이 개발되어 왔다. 이들 가공 방법을 위한 금형의 설계 및 제작은 다 품종 소량생산의 형태이면서 고도의 정밀도를 요구받으며, 제품의 라이프 사이클(life cycle)이 짧아지고 있는 것에 맞추어 단 납기화가 요구되고 있는 추세이다. 그럼에도 불구하고 대부분의 경우 실제로는 경험이 풍부한 숙련 기술자들의 경험적인 노

하우(know-how)에 의존하여 금형설계를 수행하고 있는 실정이다.

이와 같은 문제점을 해결하기 위해서 금형의 설계 및 제작에 CAD/CAM(computer-aided design/computer-aided manufacturing)시스템을 도입하게 되었다. 하지만 기존의 CAD/CAM 시스템들이 제공하는 기능들은 금형의 형상이 적합하게 만들어졌다고 가정하고 응력 분포의 계산, 냉각 및 충전 과정의 시뮬레이션, NC공구 경로의 계산 등에 초점이 맞추어져 있다. 그러므로 실제 금형 설계의 핵심이 되는 금형 캐비티부의 형상 및 치수와 이에 수반되는 금형의 부품 및 기구를 설계하는 데는 큰 도움을 줄 수 없다. 이러한 기존의 CAD/CAM 시스템의 단점을 보완하기 위해서 몇 편의 연구가 이

*서울대학교 기계설계학과 대학원

**정회원, 서울대학교 기계설계학과

미 진행된 바 있는데 먼저 이 상현⁽¹⁾은 일련의 금형 설계 과정을 대부분 CAD화 하였고 권 병옥⁽²⁾은 불리안 작업(Boolean operation)^(4~7)없이 오일러 작업(Euler operation)^(4~7)만을 이용한 새로운 방법의 코어, 캐비티 자동 생성 프로그램을 개발하였다. 그리고 허용정⁽³⁾은 사출 성형 제품의 부형상 설계를 위한 지식 베이스 CAD 시스템을 구축하였다.

그러나 앞에서 서술된 연구에서도 금형 설계의 가장 기본이라 할 수 있는 금형의 캐비티 생성 과정 중에 금형과 제품과의 언더 컷(under cut) 발생 여부에 대한 고찰이 없었다는 것이 아쉬운 점으로 지적될 수 있다. 이와 같은 금형과 제품간의 간섭 문제는 금형 설계자가 일반 모델러로 제품을 모델링할 경우, 단지 육안으로 판단할 수 없는 경우에 심각한 결과를 초래할 수 있다. 또한, 분할선(parting line)의 입력 양상에 따라라도 금형과 제품간의 간섭 부위가 크게 바뀔 수 있는 문제이기 때문에 금형 설계를 목적으로 하는 프로그램이라면 제품과 충돌하게되는 금형의 간섭면들을 자동으로 찾아내서 설계자로 하여금 수정할 수 있게 해 주는 기능이 있어야만 한다.

실제 산업 현장에서는 플라스틱 제품의 금형을 설계함에 있어 위와 같은 부품을 사용하거나, 처음부터 분할형의 금형을 설계하여 해결하고 있다. 본 연구에서는 금형을 상, 하판의 두 개로만 만든다는 가정하에 제품과 금형간에 간섭이 생길 경우 간섭면들을 찾아낸 후, 이를 이용하여 적합한 형상의 사이드 코어를 설계하는 프로그램을 개발하였다. 하지만 분할형의 금형만으로 해결되는 간섭 문제의 고려는 추후 과제로 남겨 놓았다.

Fig. 1은 전형적으로 사이드 코어가 사용되는 제품의 형상에 있어 금형의 상, 하판과 사이드 코어, 그리고 제품이 결합되어 있는 모습이다. 일반적으로 사이드 코어는 사이드 코어의 슬라이딩을 지지해 주는 슬라이드 호우더(slide holder)와 금형을 제품으로부터 분리시킬 때 사이드 코어를 바깥 방향으로 밀어내는 역할을 하는 앵글러 핀(angular pin)등과 일체로 사용된다. 이 때 사이드 코어에 코일 스프링을 부착하여 일정한 장력을 유지시켜 줌으로써 앵글러 핀과 사이드 코어의 충돌을 방지하게 된다. 사이드 코어의 작동방법은 Fig.1에서 볼 수 있듯이 금형의 상판에 붙어 있는 앵글러 핀의 경사각에 의해서 상판이 위로 분리될 경우 자동

적으로 사이드 코어가 슬라이드 호우더를 따라서 제품으로 부터 분리되는 방향으로 미끄러지게 된다.

본 연구의 내용을 요약하면 다음과 같다. 먼저 제품에 분할선⁽⁸⁾을 입력하여 상, 하판 금형을 만든 후 제품의 추출을 방해하는 금형의 면들을 간섭검사를 통하여 찾아내는 프로그램을 개발하였다. 그리고 찾아낸 간섭면들을 이용하여 금형 설계자가 적합한 형상의 사이드 코어를 설계할 수 있도록 설계과정을 CAD화 하였다. 물론, 내부적으로는 사이드 코어의 생성과 동시에 금형의 형상도 수정되도록 프로그램 되어있다. 이러한 일련의 과정은 Fig. 2와 같이 나타낼 수 있는데 앞으로 2장에서는 금형의 생성 및 간섭면 검색 과정을 다루고 3장에서는 사이드 코어의 설계 및 금형의 수정과정을 단계별로 자세히 설명하도록 하겠다. 그리고 결론에서는 현재 개발한 프로그램의 효용 및 적용 범위, 문제점 및 개선 방안 등에 대해서 논의해 보겠다.

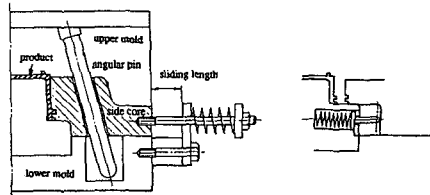


Fig. 1 Typical example of side core

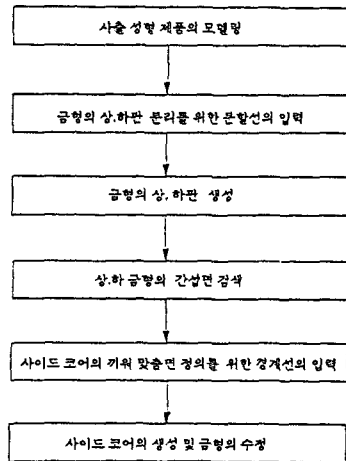


Fig. 2 Overall procedure of mold generation

2. 금형의 생성 및 간섭면 검색

2.1 금형 상, 하판의 생성

권 병욱의 연구⁽²⁾에서는 임의의 형상을 가진 3차원 물체에 대해서 사출금용형 코어판(core plate)과 캐비티판(cavity plate)의 자동 생성 문제를 다루었다. 이 방법은 Fig. 3(a)와 같이 설계자가 입력한 분할선(parting line)들을 중심으로 사출 성형 제품의 면들을 두 그룹으로 분리시킨 뒤 각각에 대해서 적당한 오일러 작업(Euler operation)을 수행하여 Fig. 3(b)와 같이 분할면(parting surface)⁽⁸⁾들을 만든 후 최종적으로 Fig. 3(c)와 같은 금형의 상판과 하판을 생성시켰다. 본 연구에서는 이 방법에 의해서 생성된 상, 하 금형을 간섭 검색의 대상으로 사용하였다.

2.2 간섭면의 검색

기존의 연구에서는 Fig. 3에서와 같이 제품자체의 형상에 간섭 부위(interference region)가 없는 경우만을 다루었다. 하지만 Fig. 4의 경우처럼 일반적인 플라스틱 제품에서는 금형의 언더 컷(under cut)을 유발시키는 간섭면들을 가지는 경우가 허다하다. 이런 경우에 간섭 부위는 크게 오목한 형상(depression feature)과 볼록한 형상(protrusion feature)의 두 가지 유형으로 분류할 수 있다. 본 연구에서는 제품의 빠짐을 방해하는 금형의 면들을 찾기 위해서 간섭 검사를 수행하였다. 이

방법에 의해 Fig. 5(b)와 같은 결과를 얻을 수 있다. 여기서 간섭 검색의 대상이 되는 면들은 Fig. 5(a)와 같이 금형 상판의 외곽면을 제외한 모든 면들이 된다.

여기서는 금형의 상판과 하판의 간섭 검사 방법이 대칭적으로 같은 원리를 사용하기 때문에 상판의 면들을 예로들어 설명하겠다.

(1) 1차 간섭면의 탐색

금형의 개폐 방향을 Z방향이라하면 상판의 면들 중 Fig. 6(a)의 왼쪽 평면과 같이 면의 바깥 쪽 방향 법선 벡터(normal vector)가 +Z성분을 가지는 것들이 제일 먼저 발견할 수 있는 간섭면들이다. 왜냐하면 금형에서 제품이 빠지는 방향은 -Z방향이기 때문이다. 이와같이 면의 법선 벡터를 검사하는 방법은 면의 곡면식이 평면일 경우는 평면위의 임의의 한 점만의 법선 벡터 검사로 가능하며, 평면이 아닌 일반적인 자유 곡면의 경우는 u,v매개변수(parameter)를 각각 nseg만큼 등분하여 nseg x nseg개의 시험점의 법선 벡터를 검사하여 판단할 수 있다. 자유곡면의 경우는 Fig. 6(a)의 오른쪽 곡면과 같이 곡면 자체에 +Z와 -Z성분의 법선 벡터를 가지는 점들이 공존하는 경우가 생기는데, 이러한 경우에는 Z축 위에서 곡면을 보았을 때 실루엣 커브(silhouette curve)⁽⁹⁾가 생기기 때문에 이



Fig. 4 Regions of product causing interference

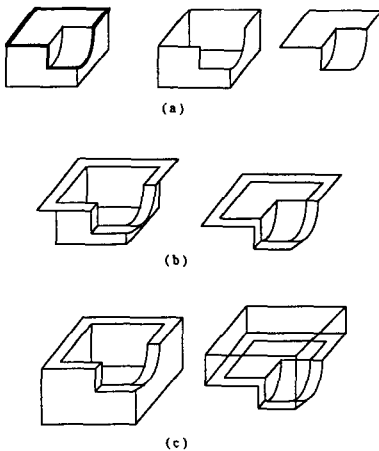


Fig. 3 Automatic generation of cavity and core plate

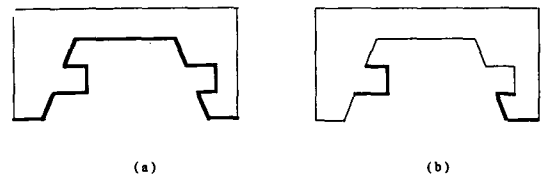


Fig. 5 Identification of faces of mold causing interference

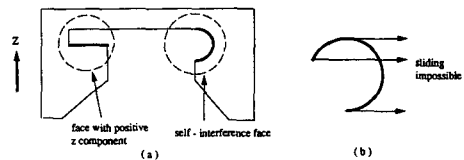


Fig. 6 Primary interfering faces

러한 실루엣 커브를 구해서 그 갯수 만큼 면을 분할시켜 주게 된다. 이러한 면의 분할작업으로 간섭에 기여하는 부분만을 새로운 간섭면으로 찾아낼 수 있으며 사이드 코어를 설계할 경우 사이드 코어의 빠짐을 불가능하게 하는 Fig. 6(b)와 같은 경우를 제거해주게 된다. 만약 면의 법선 벡터 검사 방법에 의해서 간섭면이 나타나지 않았다면 금형의 전체 형상에 간섭 부위가 존재하지 않는다는 결론을 내릴 수 있다.

(2) 2차 간섭면의 탐색

Fig. 7(a)에서 제품의 면중 금형과 간섭을 일으키는 면은 face1, face2, face3이고 이들 면들이 아래로 빠지는 것을 방해하는 금형의 면들은 1차 간섭면인 face4뿐 아니라 face5, face6임을 한눈에 알 수 있다. 이 사실을 기본 원리로 하여 모든 제품의 면에 대해서 모든 금형의 면들에 대한 간섭 검사가 필요하다. 하지만 앞절의 1차 간섭면의 탐색에서 한 가지 중요한 사실을 유도해 낼 수 있다. 즉 금형의 간섭 부위에는 최소한 하나 이상의 1차 간섭면들이 존재한다는 사실이다. 결국 이 말은 1차 간섭면들에 의해서 제품과 금형 사이에 간섭 현상이 일어난다는 뜻으로 풀이할 수 있다. 이 사실을 근거로하여 Fig. 7(b)에서는 face1, face2, face3 각각의 면이 face4, face5, face6에 의해서 간섭받는 영역의 합집합인 face3이 face4, face5, face6에 의해서 간섭받는 영역과 일치 한다는 것을 알 수 있다. 이때 제품의 면인 face3는 금형의 면 중 1차 간섭면인 face4와 동일한 면이다. 그러므로 face4를 이용하여 face4가 아래로 빠질때 방해되는 face5, face6를 2차 간섭면으로 검색해냄으로써 금형의 모든 간섭면들을 찾아낼 수 있다. 결론적으로 본 연구에서는 제품의 면은 고려하지 않고 금형의 면 중에서 1차 간섭면들을 이용하여 1차 간섭면의 움직임을 방해하는 2차 간섭면을 검색해냄으로써 모든 간섭면들을 찾아낼 수 있었고, 이러한 간단한 원리를 이용하여 간섭 검사의 전 과정을 수행하기 때문에 아무리 복잡한 형상의 금형일지라도 단 시간내에 간섭면들을 찾아낼 수 있는 장점이 있다.

앞에서 소개된 원리를 이용한 2차 간섭 검사의 전 과정을 요약하면 아래의 몇 단계로 설명할 수 있다.

첫째, 면의 법선 벡터 검사에 의해서 찾아낸 1차 간섭면들 중 자체 간섭면(self-interference face)들을 가려내서 각각에 대해서 실루엣 커브를 찾는다.

(이 과정의 설명은 부록에 자세히 소개되어 있다.) 그리고 찾아낸 실루엣 커브를 이용하여 Fig. 8과 같이 오일러 작업을 통해서 원래의 면을 간섭면(interference face)과 비간섭면(non-interference face)의 두 개로 분리시킨다.

둘째, Fig. 9(a)와 같이 모든 면들에 대해서 경계 모서리(boundary edge)를 따라 가면서 한 방향을 가지면 직선들의 집합으로 이루어진 근사 다각형(approximated polygon)을 만든다. 이때 직선인 경우는 2개의 점으로 정확하게 나타낼 수 있으며 자유 곡선의 경우는 매개변수 t에 대하여 nseg 만큼 등분하여 nseg개의 근사적인 직선으로 나타내게 된다. 이러한 일련의 과정은 간섭 검사를 위한 근사 다각형 거리의 2차원 교점 계산 및 In-Out

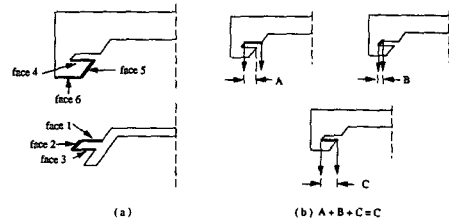


Fig. 7 Mold faces preventing motion of product faces

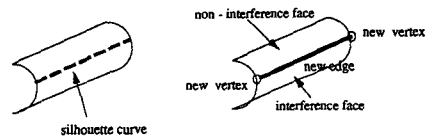


Fig. 8 Separation of self-interference face at silhouette curve

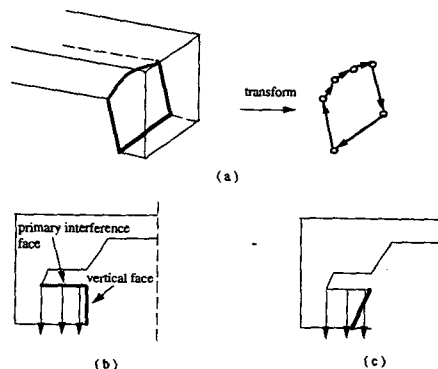


Fig. 9 Conversion of faces for easy interference checking

검사를 하기 위한 준비 작업이다. 하지만 수직 한 면(vertical face)의 경우는 Fig. 9(b)에서 볼 수 있듯이 면의 근사 다각형으로만 간섭 검사를 할 경우 감지되지 않는 경우가 발생하기 때문에 Fig. 9(c)에서와 같이 금형의 부피가 줄어드는 수평 방향으로 면에 구배각(draft)을 적용하여 경사진 근사 다각형을 만들며, 아울러 원래의 곡면과는 다른 경사진 경계에 맞는 새로운 곡면을 생성하여 따로 갖고 있게 된다. 이 곡면은 나중 단계에서 무한 직선과의 교점 계산을 위한 곡면으로 사용된다.

세째, 1차 간섭면의 움직임을 방해하는 2차 간섭면을 찾는 과정이다. 1차 간섭면을 face i라고 하고 간섭 검사의 대상면을 face j라고 하면, 모든 face i에 대해서 모든 face j를 아래의 몇 과정으로 써 검사하게 된다.

●과정 1. face i와 face j의 근사 다각형을 XY 평면에 투사(projection)시킨 후 이들의 2차원 비교 사각형(minimum-maximum square)이 겹치면 다음 과정으로 넘어 간다. 그렇지 않으면 j를 1만큼 증가 시키고 다시 검사를 한다.

●과정 2. Fig. 10(a)에서와 같이 face i와 face j의 근사 다각형을 XY평면에 투사시켜 교점을 계산한다. 이 과정은 Fig. 10(b)에서와 같이 실제로는 2차원 평면 상에서의 직선과 직선의 교점 계산으로 볼 수 있다. 이렇게 하여 2개의 교점이 나오면 그 교점의 X,Y값으로부터 투사되기 전의 근사 다각형상의 해당점을 구하여 이들의 Z값을 비교하여 $pt1.z > pt3.z$ 이고 $pt2.z > pt4.z$ 이면 pt1과 pt2 사이에 있는 근사 다각형의 꼭지점들(만약 pt1과 pt2 사이에 꼭지점이 없으면 두 값의 평균값)인 vertex 1, ..., vertex n중 한 점이라도 face j의 근사 다각형 안에 있으면 face j는 face i의 2차 간섭면으로 판명된다. 아니면 계속해서 2개의 교점이 나올때 까지 프로그램을 수행하게 된다. 최종적으로도 간섭면으로 판명되지 않으면 다음 과정으로 넘어가게 된다. 이와 같은 방법은 3차원 공간 상에서는 금형의 면들간에 면의 경계를 제외하고는 내부적으로 서로 교차하는 부분이 없기 때문에 가능하다.

●과정 3. 이 과정에서는 근사 다각형 끼리의 교점 계산 방법으로 검사할 수 없는 Fig. 11과 같은 경우들을 검사하게 된다. Fig. 11(a)는 face i의 근사 다각형이 face j의 근사 다각형 안에 포함되는 경우인데 이 때는 face i의 경계안에 있는 임의의 한점으로 부터 아래로(금형 하판의 경우는 위

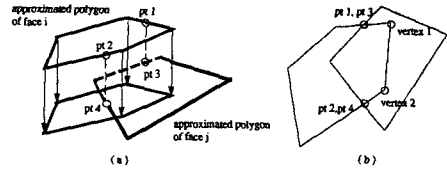


Fig. 10 Calculation of intersection points between approximated polygons

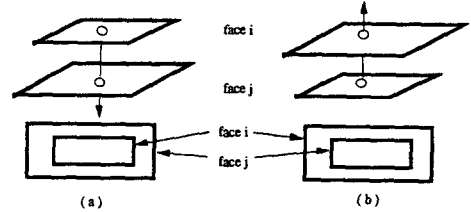


Fig. 11 Interference faces without intersection points between approximated polygons

로) 직선을 쏘아서 face j의 곡면과 교점을 구하게 된다. 만약 교점이 있고 그 점이 face j의 근사 다각형 안에 있다면 face j는 2차 간섭면으로 판정된다. 그렇지 않다면 Fig. 11(b)에서와 같이 face i의 근사 다각형이 face j의 근사 다각형을 포함하는 경우를 검사하게 된다. 이 과정은 앞의 과정을 역으로 수행하게 된다. 여기서도 face j가 간섭면으로 판명되지 않으면 face j는 비간섭면으로 분류된다. 이러한 일련의 과정에서 곡면 위의 임의의 한점을 계산하기 위해서는 곡면의 매개 변수 u, v 값을 변화 시켜가면서 구한 곡면 위의 점이 면의 근사 다각형 안에 있을 때 그 점을 선택하게 된다.

3. 사이드 코어의 설계 및 금형 수정의 CAD화

본 연구에서의 사이드 코어 설계 과정은 많은 제한과 단순화가 이루어 졌기 때문에 앞으로 기술하게 될 일련의 과정들이 실제 현장에서 근무하는 금형 기술자들의 설계 방법과는 많은 차이점을 보일 지도 모른다. 그러나 자료 분석과 많은 고찰을 통해서 가급적이면 프로그램 사용자가 직접적으로 설계 데이터를 통해서 설계 과정에 접근할 수 있도록 프로그램 하였다. 사이드 코어가 생성되기 까지의 과정을 간단히 요약하면 아래의 몇 단계로 분류할 수 있다.

첫째, 찾아낸 간섭부위 중에서 그것 자체 만으로도 적당한 모양의 사이드 코어를 만들어 낼 수 있는 경우도 있다. 하지만 사이드 코어의 설계에 있어 간섭면들의 역할은 이 간섭면들이 사이드 코어의 끼워 맞춤부의 형상에 반드시 나타나야 한다는 것 뿐이다. 실제로 Fig. 12(a)에서와 같은 간섭면에 대해 Fig. 12(b)와 같이 사이드 코어의 전체 형상은 설계자의 의도 뿐만 아니라 금형의 전체 형상에 따라서 매우 다양한 모양을 갖고 있는 것이 보통이다. 그래서 본 연구에서는 설계자가 직접 사이드 코어의 끼워 맞춤부에 해당하는 면들을 지정할 수 있도록 찾아낸 간섭면들로 부터 경계선을 입력하게 했다.

둘째, 사이드 코어의 슬라이딩 방향(sliding direction)을 입력 한다. 이 과정은 Fig. 13(a)와 같이 설계자가 사이드 코어가 제품으로 부터 이탈하는 방향을 직접 입력하게 했다. 여기서 설계자가 입력한 방향으로 장차 사이드 코어의 끼워 맞춤면들이 될 금형 면들의 법선벡터를 검사하여 사이드 코어의 빼냄이 가능한가를 확인한다. 만약 Fig. 13(b)와 같이 사이드 코어의 빼냄이 불가능한 경우에는 슬라이딩 방향을 다시 입력하게 된다.

세째, 경계선에 의해 분리된 끼워 맞춤면들에 대

해 적당한 오일러 작업을 행하여 사이드 코어의 몸체 부분을 완성한다. 한편 끼워 맞춤면을 제외한 금형의 면에도 똑같은 작업을 수행하여 금형의 형상을 수정하게 된다. 이 과정에서는 설계자가 사이드 코어의 전체 크기를 고려하여 적절한 데이터를 입력할 수 있도록 하였다.

위의 각 단계에 대한 자세한 설명은 다음 각 절에서 소개한다.

3.1 사이드 코어의 끼워 맞춤면 분리

금형 설계자는 사이드 코어의 끼워 맞춤면을 정의해 주기 위해서 아래의 Fig. 14(a)에서와 같이 끼워 맞춤면의 경계선을 입력하게 된다. 이 때 프로그램에서는 입력된 경계선 중심으로 Fig. 14(b), (c)에서와 같이 금형의 면들을 사이드 코어에 붙게 될 면과 금형의 형상에 계속적으로 남아 있게 될 면 등의 두 그룹으로 분류한다.

이 때 Fig. 15에서와 같이 입력한 경계선과 사이드 코어의 슬라이딩 방향에 있는 금형 외측면의 경계 모서리를 슬라이딩 방향으로 보았을 때 중첩되는 부분이 있어야 하며, 중첩되는 부분의 폭이 분할선을 완전히 포함해야 한다. 그렇지 않으면 나중에 사이드 코어와 금형의 간섭을 일으키는 현상이 발생하게 된다.

한 입체에서 면들을 두 개의 그룹으로 분리하는 과정은 그 입체를 복제한 입체로 부터 각각 반대편

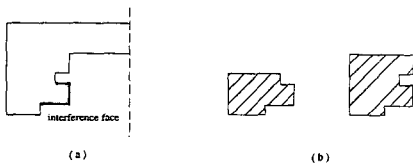


Fig. 12 Various shapes of side cores

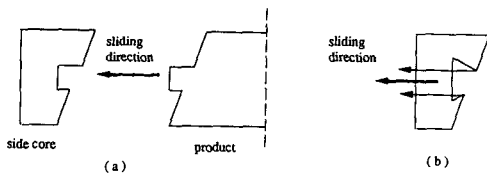


Fig. 13 Sliding direction of side core

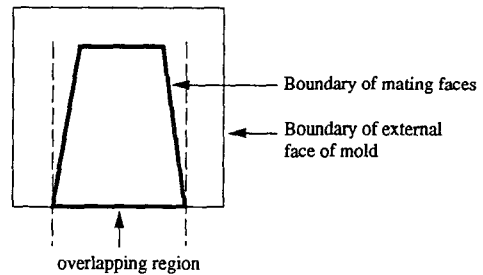


Fig. 15 Requirement on boundary lines

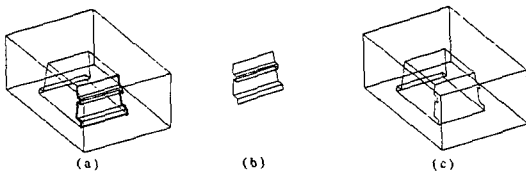


Fig. 14 Input of boundary lines of side core

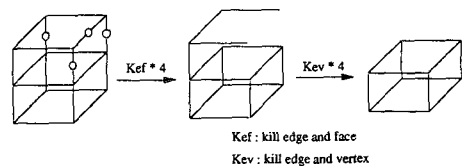


Fig. 16 Elimination of faces in opposite group

그룹의 면들을 제거해 내면 되므로 Fig. 16과 같은 면의 제거 과정을 거치면 된다. 입체로부터 면의 집합을 제거하는 것은 분할선에 있는 모서리들을 제외하고 그 면에 속한 모든 모서리들을 제거하는 것과 동일하다. 따라서 면의 집합으로부터 제거되어야 할 모서리들을 찾은 후 적당한 오일러 작업에 의해 각각의 모서리를 제거하게 된다.

위에 설명한 방법으로 원래 금형의 데이터를 하나 복제하여 Fig. 16과 같은 면의 제거 과정을 거쳐서 Fig. 14(b)와 같은 모양의 사이드 코어 데이터를 만든다. 또한 원래 금형에서 사이드 코어의 면들을 제거하여 Fig. 14(c)와 같은 모양으로 금형의 데이터를 수정하게 된다.

3.2 사이드 코어의 외측면 생성

사이드 코어의 외측면 역시 오일러 작업을 통하여 원하는 모양으로의 구현이 가능하다. Fig. 17(a)에는 금형으로부터 복제되어 면의 제거 과정을 거쳐 생성된 사이드 코어의 끼워맞춤면들이 표시되어 있다. 일반적인 방법은 Fig. 17(b)와 같이 사이드 코어의 경계선 위에 있는 꼭지점들을 따라서 사이드 코어의 슬라이딩 방향으로 먼저 Mev(make edge and vertex)를 수행하게 된다. 이 때 슬라이딩 방향에서 보아서 겹치게 되는 꼭지점들은 슬라이딩 방향으로의 무한한 점에서 가장 가까운 꼭지점에 대해서만 Mev를 수행하게 된다. 이 원리에 의해서 Fig. 17(b)의 vertex 1은 Mev를 수행하지 않았다. 그리고 vertex 2와 같이 슬라이딩 방향에서 보았을 때 한 직선상에 놓이는 꼭지점에서는 Mev를 수행하지 않는다. Mev과정이 끝나면 Fig. 17(c)와 같이 새로 생긴 꼭지점들을 Mef(make edge and face)로 모두 이어서 새로운 면들을 만들어 준다. 이때 새로 생긴 각 꼭지점을 잇는 곡선은 분할선상에 있는 곡선들을 슬라이딩 방향으로 투사(projection)시켜서 만들게 된다.

이와 같은 사이드 코어의 경계선을 리프팅하는 방법은 설계자가 원한다면 경계선 내에 새로운 모서리를 생성하여 경계선의 일부만을 리프팅할 수도 있다. 그런데 Fig. 18(a)와 같은 경우는 경계선으로부터 Fig. 17과 같은 방법으로 사이드 코어를 생성할 경우 박판(sheet)이 생기거나, 형상적으로 날카로워서 매우 취약한 부분을 발생시키게 된다. 이때 박판이 생기는 지의 판단을 위해서는 Mev수행 당시 새로 생길 모서리가 경계선에 의해 1차적

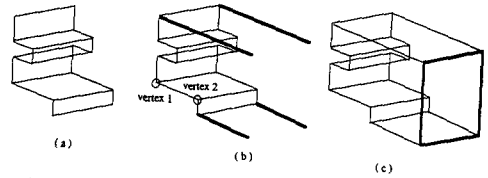


Fig. 17 Generation of side core from boundary lines using Mev & Mef

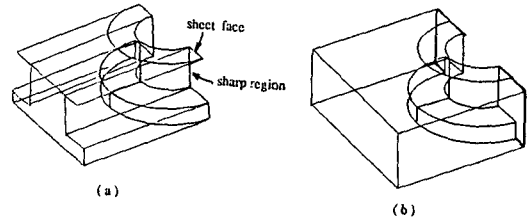


Fig. 18 Extension of boundary of mating faces

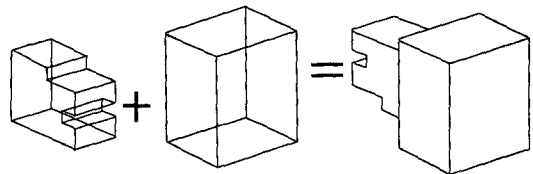


Fig. 19 Completion of side core

으로 분리되어 생성된 사이드 코어의 면들과 접하는 지 아닌지를 검사하여 확인할 수 있다. 이와 같은 문제를 해결하기 위해서 본 연구에서는 Fig. 18(b)와 같이 슬라이딩 방향에 수직인 방향으로 접합면의 확장이 가능하도록 하였다. 하지만 접합면의 확장될 부분이 평면이 아닌 경우는 나중에 사이드코어가 금형과 간섭을 일으키기 때문에 확장이 불가능하다.

최종적으로는 윗단계에서 만들어진 사이드 코어의 형상이다, 사이드 코어의 전체 크기를 설계자로부터 입력 받아서 만든 직육면체(box)를 Fig. 19와 같이 접합시키게 된다. 여기서 사이드 코어에 접합시킨 직육면체에는 사이드 코어의 슬라이딩 거리를 고려한, 앵글러 핀이 꽂힐 경사진 통로를 만들게 된다. (본 연구에서는 다루지 않았음)

3.3 금형 형상의 수정

금형 형상의 수정은 사이드 코어의 생성과 동시에 Fig. 20과 같은 상판의 수정 및 Fig. 21과 같은 하판의 수정과정으로 이루어진다. 결과적으로 보면 앞의 과정에서 생성된 사이드 코어를 원래의 상,

하 금형에서 불리안 작업을 통해서 빼낸 것과 같은 효과를 내고 있다. 하지만 내부적으로는 오일러 작업만을 사용하기 때문에 불리안 작업을 사용하는 것과 비교하여 훨씬 빠르고 정확하다.

먼저 금형 상판의 수정과정을 살펴보면, Fig. 20 (a)는 권 병옥의 알고리즘에 의해 최초로 만들어진 금형 상판이다. 그리고 Fig. 20 (b)는 사이드 코어의 끼워 맞춤면을 분리시킨 뒤 1차 수정된 금형 상판의 모습이다. Fig. 20 (c)는 슬라이딩 방향에 있는 금형 외측면에 사이드 코어의 직육면체와 중첩되는 부분 만큼 새로운 면을 만든 것이다. Fig. 20 (d)는 사이드 코어의 외측면 생성시 사용했던

리프팅 기법을 사용하여 금형의 경계선을 리프팅 시킨 것이다. 이때 경계선과 사이드 코어의 슬라이딩 방향에 있는 금형 외측면의 경계 모서리를 슬라이딩 방향에서 보았을 때 중첩되는 부분은 리프팅 시키지 않는다. Fig. 20 (e)는 Fig. 20 (d)에서 리프팅 된 루프(loop)를 사이드 코어의 직육면체에 해당하는 공동부를 만들기 위해 오일러 작업으로 연장시킨 것이다. Fig. 20 (f)는 Fig. 20 (d)에서 리프팅하지 않은 중첩부를 제거한 것이다. Fig. 20 (g)는 금형 외측면에서 아래에 있는 모서리(base edge)들을 제거한 것이다. 최종적으로 Fig. 20 (h)와 같이 금형 외측면의 꼭지점과 경계선의 연장상에 있는 꼭지점을 Mef로 이어서 Fig. 20 (i)와 같은 금형 상판을 완성한다.

금형 하판에 대해서도 앞에서 설명한 것과 유사한 방법으로 수정과정을 거쳐 Fig. 21 (e)와 같은 최종 형상을 얻게 된다.

4. 적용 예

Fig. 22, 23은 각각 간섭 검사를 통하여 사이드 코어를 생성한 예들이다. 각 그림의 (a)에는 제품에 대한 1차적인 금형 형상과 제품의 추출을 방해하는 금형면들을 간섭검사에 의해 찾아낸 결과를 함께 표시해 놓았다. 또한 그림에서 A부분은 금형 생성에 이용된 분할선을 나타내며 B부분은 사이드 코어의 생성에 사용된 경계선을 나타낸다. 이들 분할선과 경계선은 사용자에게 의해 대화식으로 입력된다. Fig. 23 (b)에서는 사이드 코어의 경계선에 의해 분리된 접합면들을 Fig. 18 (b)와 같이 확장해서 사이드 코어를 생성한 것이다. 한편 각 그림의 (b)는 Fig. 20과 21의 과정을 거쳐서 최종적으로 생성한 사이드 코어들과 수정된 금형 형상을 함께 제시해 놓았다.

5. 결론 및 제안

본 연구에서는 금형과 제품간의 언더컷 문제를 해결하는 한 방법으로 간섭면 검색을 통해 사이드 코어를 설계할 수 있는 방법을 개발 하였다. 이것은 기존의 연구에서는 다루어지지 않았던 문제로서, 실제 산업 현장에서는 금형 설계를 함에 있어 가정 먼저 고려해야 할 중요한 문제라는 점에서 그 의미를 찾을 수 있다. 특히 간섭면의 검색 과정은

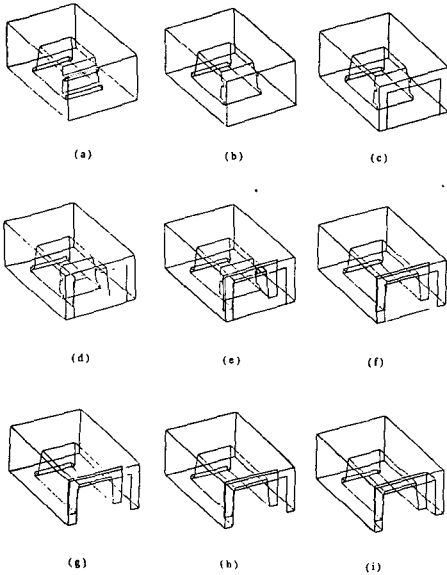


Fig. 20 Modification of upper mold

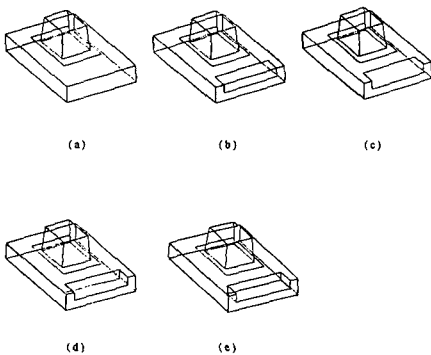


Fig. 21 Modification of lower mold

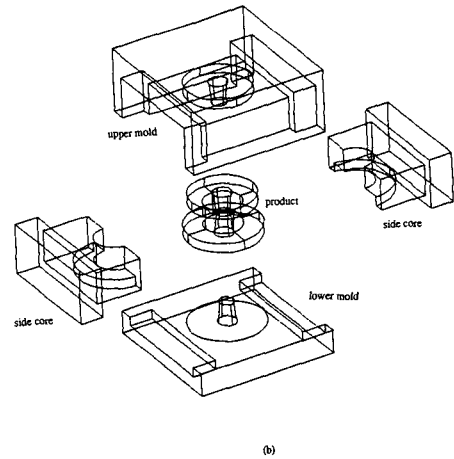
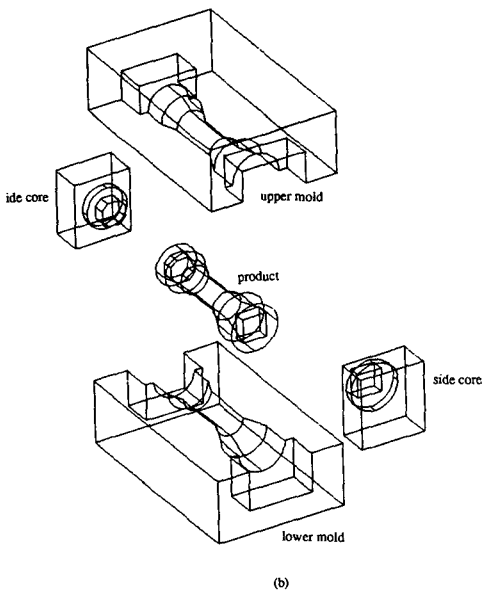
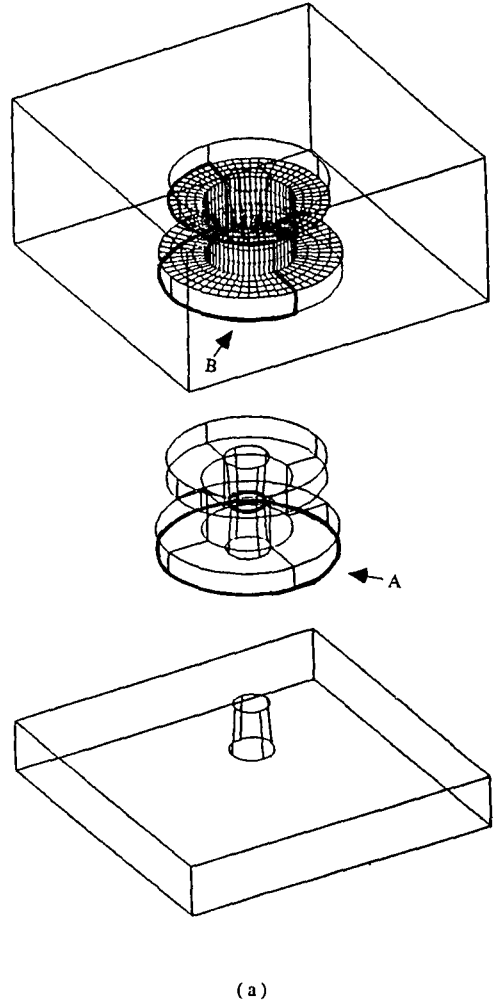
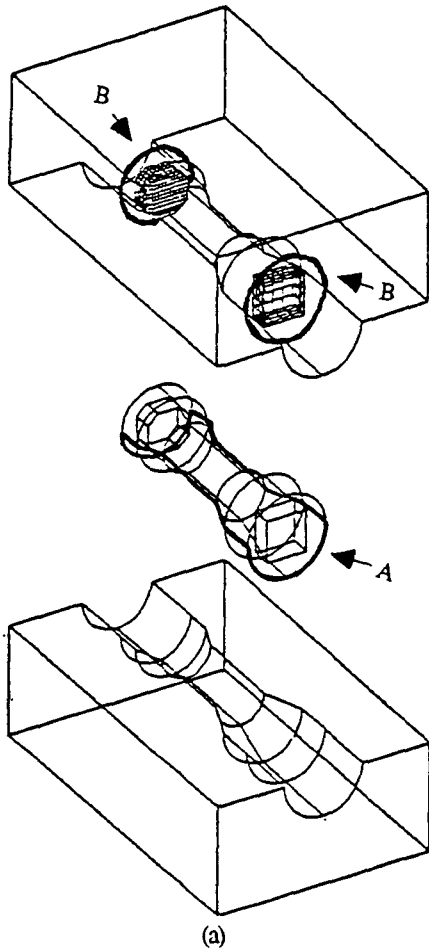


Fig. 22 Generation of side core in a typical case

Fig. 23 Generation of side core by expanding mating faces

모든 경우에 대해 일반적으로 적용할 수 있기 때문에 분리 방향만 주어지면 사출 금형은 물론, 주물 및 그 밖의 끼워 맞춤면이 존재하는 조립체에도 적용이 가능하다.

한편 현재까지 개발된 사이드 코어의 생성 기능을 이용하여 Fig. 1과 같은 간섭 문제는 물론 Fig. 22, 23과 같이 좀더 복잡한 간섭현상도 해결 할 수 있다. 하지만 이 방법으로 모든 간섭 문제를 해결하기에는 몇가지 미흡한 점이 있다. 먼저 Fig. 23 (b)의 경우처럼 입력한 경계선을 확장해서 사이드 코어를 생성하여야 하는 경우에 접합면 확장 방법의 한계(슬라이딩 방향에 대해 수직인 평면으로 확장)로 인해 설계자가 원하는 모든 형상을 충분히 만족 시킬 수는 없다. 그리고 똑같은 간섭 문제를 놓고 사이드 코어를 쓰기 보다는 분할형의 금형 설계가 편리한 경우가 있을 수 있으므로 분할형의 금형 설계도 지원될 필요가 있다. 또한 접합면을 정의하는 경계선의 입력을 제품의 형상이 복잡한 경우에는 설계자가 부담을 줄 수 있으므로 개선이 필요하다. 이외에도 금형의 다른 부품(런너, 게이트, 냉각회로, 밀어내기 기구), 제품, 그리고 금형간의 간섭 문제도 추후에는 고려될 필요가 있다.

이상에서 살펴본 몇가지 문제가 해결된다면 본 연구의 내용은 금형 설계를 위한 기초적인 수단으로 유용하게 쓰일 수 있을 것이다.

참고 문헌

- (1) Lee, S.H. and Lee, K., 1988, "An Integrated CAD System for Mold Design in Injection Molding Processes," 1988 ASME Winter Annual Meeting, Chicago, pp. 257~271.
- (2) 권병욱, 이건우, 1991, "사출 금형 설계를 위한 코어와 캐비티의 자동 생성", 대한 기계학회 논문집 제 15권 제 4호, pp. 1225~1232
- (3) 허용정, 1991, "사출 성형의 제품의 부 형상 설계를 위한 지식형 CAD시스템에 관한 연구", 박사학위논문, 한국과학기술원 기계공학과.
- (4) Hiroaki Chiyokura, 1988, "Solid Modeling with DESIGNBASE," Addison-Wesley.
- (5) Matti Mantyla, 1988, "Solid Modeling," Comuter Science Press.
- (6) 1983, "ROMULUS Version 5.0 Tutorial Users Guide".

- (7) 박상호, 1990, "특징 형상을 입력으로 하는 모델러의 개발", 석사학위논문, 서울대학교 기계설계 학과, pp. 24~60.
- (8) Ravi, B. and Srinivasan, MN, 1990, "Decision Criteria for Computer-aided Parting Surface design," Computer-Aided Design Vol. 22 No. 1.
- (9) Michael E. Mortenson, 1985, "Geometric Modeling," John Wiley & Sons, pp. 533~537.
- (10) William, H. Press., Brian, P. Elannery, Saul A. Teukolsky, William T. Vetterling, 1988, "Numerical Recipes in C," Cambridge University Press, pp. 261~263.

부록

● 실루엣 커브의 정의 및 계산

앞에서 소개된 실루엣 커브(silhouette curve)의 정의 및 계산 방법에 대해서 간단히 소개하면 아래와 같다. 실루엣 커브는 Fig. 24와 같이 $\vec{z}\vec{n}=0$ 을 만족하는 점들의 집합이다. 여기서 \vec{z} 는 Z방향으로의 벡터, 즉 (0,0,1)이고 \vec{n} 는 곡면의 바깥쪽 법선 벡터이다.

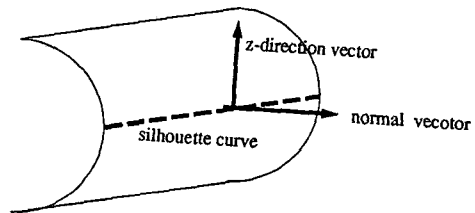
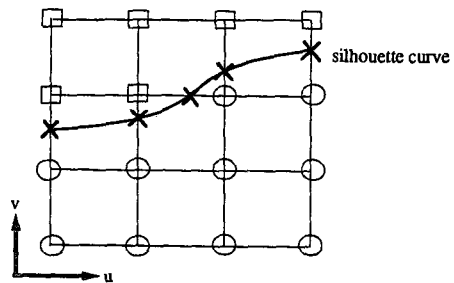


Fig. 24 Definition of silhouette curve



- : points with +z component of normal vector
- : points with -z component of normal vector
- × : points without z component of normal vector

Fig. 25 Calculation of silhouette curve

먼저 곡면의 매개 변수(parameter) u, v 를 입력된 수($nseg$)만큼 등분한 후 모든 등분점에서의 법선 벡터와 Z 방향 벡터(0,0,1)의 내적(dot product)을 구하고 이들을 $nseg \times nseg$ 의 배열로 저장한다. 이들 배열에 저장된 값들의 부호만을 가지고 Fig. 25에 보인 것처럼 실루엣 커브가

어디를 지나는지 알 수 있다.

즉 한 등분점과 인접한 등분점들 간의 부호가 다르다면 그 두 점 사이에서 법선 벡터와 Z 방향 벡터의 내적이 영이 되는 점이 존재한다. 그런 점들은 바이섹션법(Bisection Method)⁽¹⁰⁾에 의해서 찾을 수 있다.