

바둑판의 회전과 대칭 처리를 위한 WYSWYG 인터페이스

양단희
평택대학교 컴퓨터학과
dhyang@ptu.ac.kr

An WYSWYG Interface for the Rotational and Symmetrical Operation of Goban

Dan-Hee Yang
Dept. of Computer Science & Engineering, Pyeongtaek University

요 약

바둑 기보를 보거나 편집하기 위한 소프트웨어들은 대부분 비슷한 인터페이스와 기능을 제공하고 있으며 별다른 성과와 기능이 더 필요하지 않을 만큼 잘 개발되어 있다. 그러나 이런 류의 소프트웨어에서 사용자가 기보를 원하는 방향에서 보기 위해 바둑판을 회전/대칭 처리하고자 할 때 매우 불편함을 경험하게 된다.

그래서 본 연구에서는 이에 대한 직관적인 인터페이스 방법을 제안하고 이를 구현하기 위한 수학적 연산을 제시하고자 한다. 본 연구에서 제안한 WYSWYG 방식의 인터페이스는 바둑 애호가뿐만 아니라 프로 기사들에게도 매우 편리한 기능이 될 것임에 틀림없다.

1. 서 론

바둑 기보를 컴퓨터를 이용하여 편리하게 복기해볼 수 있는 기능을 제공하는 기보 보기/편집 소프트웨어들이 많이 개발되어 있고, 각종 인터넷 바둑 사이트에서는 자기 고유의 기보 보기 시스템을 제공하고 있다.

이들 소프트웨어에는 삼본데이터시스템에서 개발한 삼본기보뷰어 Ver 1.00처럼 기보 복기의 기본 기능에 충실하여 한수 전/후진, 놓아보기, 자동 복기만을 제공하는 순수 기보 보기 프로그램부터 다양한 기보 편집 기능과 더불어 바둑판(기보)을 회전시키거나 x축 대칭 이동, y축 대칭 이동 등을 제공하는 MultiGo 4.2.1과 같은 다기능 기보 편집 프로그램도 있다[1].

지금까지 컴퓨터 바둑에 관한 연구는 정석 선택, 형세 판단, 다음 착수 후보 생성, 사활 문제 풀이, 끝내기 등 주로 컴퓨터의 기력을 향상 시키는 방향으로 연구가 집중되어져 왔다[2,3,4,5,6].

그리고 사용자 인터페이스와 기보 보기 시스템에 관해서는 매우 간헐적으로 연구가 수행되어 별다른 진보를 보이지 않고 있다. 그 주된 이유는 기보 관련 소프트웨어들이 별다른 기능과 성능이 더 필요하지 않을 만큼 잘 개발되어져 있기 때문일 것이다[7,8,9].

그러나 한가지 아쉬운 점은 이들 소프트웨어에서 사용

자가 기보를 원하는 방향에서 보기 위해 바둑판을 회전/대칭 처리할 때 상당한 불편함을 느끼게 한다는 것이다.

그래서 본 연구에서는 이에 대한 직관적인 인터페이스 방법을 제안하고 이를 구현하기 위한 수학적 연산을 제시하고자 한다.

2. MultiGo와 문제점

그림 1의 MultiGo는 Rui Jiang에 의해 개발된 것으로 기보 보기/편집 소프트웨어 중에서 가장 좋은 평가를 받고 있으며 Freeware이면서 매년 조금씩 성능과 기능을 꾸준히 개선 시키고 있다.

본 장에서는 MultiGo에 대해 간단히 소개하고, 바둑판의 회전/대칭 이동을 위한 인터페이스의 불편함에 대해 언급하겠다.

2.1. MultiGo의 소개

MultiGo는 기본적으로 Smart-Go Format(SGF)¹의 바둑 게임 파일을 보거나 편집하기 위한 프로그램이다. 그러나 *.BDX, *.NGF, *.GOS, *.GIB, *.UGF, *.GO와 같이 많이 사용되고 있는 다른 형식의 바둑 게임 파일도 읽어

¹ Game Tree를 저장하기 위한 표준 파일 형식

들일 수 있다. 다만 저장은 SGF로만 할 수 있도록 제한되어 있는 것이 아쉬운 부분이다.

그리고 게임과 브랜치(branch)²를 추적해 가고, 표시자(marker)나 주석(comment)을 달거나 지울 수 있고, 사용자가 자신이 브랜치를 새로 만들거나 편집할 수도 있다.

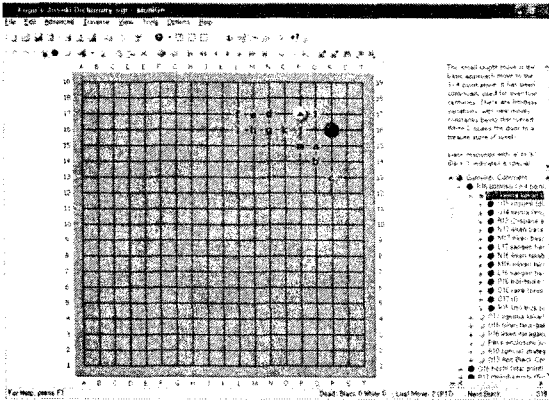


그림 1 MultiGo의 작동 화면

그림 1의 [View] 메뉴에는 바둑판에 적용할 수 있는 연산과 관련하여 그림 2와 같이 [시계 방향으로 회전], [수평으로 뒤집기], [수직으로 뒤집기], [원 상태로]를 제공하고 있다.

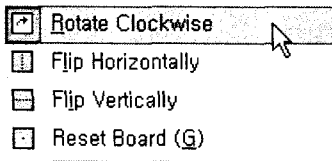


그림 2 [View]의 회전/대칭 연산 메뉴

2.2. MultiGo의 문제점

바둑판에서 우상 소목에 있는 흑돌 A에 대해 적용시킬 수 있는 회전/대칭 이동의 결과는 그림 3에서와 같이 흑돌 B, C, D, E, F, G, H의 위치가 된다.

이를 일반화시키기 위해 바둑판의 영역을 그림 3에서와 같이 빨강색 A-H까지 구분시키겠다. 그리고 A 영역을 기준으로 A 영역에 있는 한 돌을 다른 영역으로 회전/대칭 이동시키기 위해서는 표 1에서와 같은 연산을 적용시키면

된다. 여기서 회전은 수학적인 반시계 방향의 회전을 의미한다.

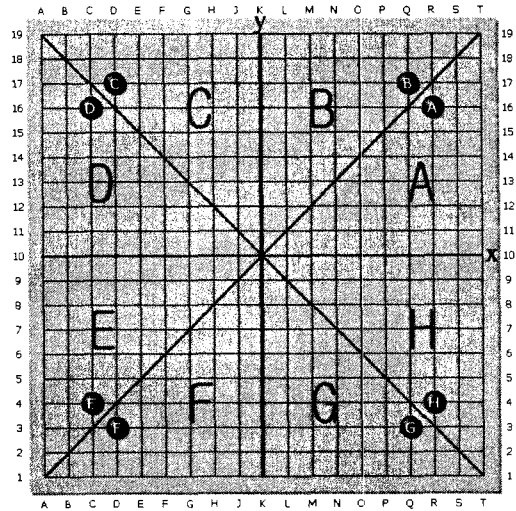


그림 3 바둑판의 영역 구분과 가능한 회전/대칭

표 1 바둑판 회전/대칭 연산

영역	연산	동치 연산
A	-	-
B	y=x 대칭	x축 대칭 후 90° 회전
C	90° 회전	-270° 회전
D	y축 대칭	-
E	180° 회전	◇ y=-x 대칭 후 y=x 대칭 ◇ 원점 대칭
F	y=-x 대칭	y축 대칭 후 90° 회전
G	270° 회전	-90° 회전
H	x축 대칭	-

예를 들어 A 영역에 있는 돌을 B 영역으로 이동시키기 위해서는 직선 y=x에 대해 대칭 연산을 수행해야 하고, C 영역으로 옮기기 위해서는 반시계 방향으로 90° 회전 연산을 수행해야 한다. F 영역으로 옮기기 위해서는 직선 y=-x에 대해 대칭 연산을 수행해야 한다.

² Tree 구조에서 branch를 말한다.

그런데 MultiGo에서는 $y=x$, $y=-x$ 에 대한 대칭 연산을 지원하지 않기 때문에 표 1의 동치 연산에 따라 $y=x$ 에 대한 대칭은 x 축 대칭 연산 후 90° 회전을 해야 하고, $y=-x$ 대칭은 y 축 대칭 연산 후 90° 회전을 해야 한다. 이는 매우 수학적 연산으로 바둑을 두는 일반인이 손쉽게 할 수 있는 성질의 것이 아니다. 이를 해결하기 위해 직선 $y=x$ 와 $y=-x$ 대칭 이동을 각각 할 수 있도록 메뉴를 제공하는 것도 하나의 해결책이 될 수 있다

또한 수학적인 회전 연산은 반시계 방향을 기본으로 하고 있는데 반해 MultiGo가 제공하는 회전은 시계방향으로 회전을 강제하고 있어 사용자에게 혼란을 야기시킬 수 있다. 또한 270° 회전은 -90° 회전을 하도록 하는 것이 훨씬 편리하지만 이 기능 또한 제공되지 않고 있다. 이를 해결하기 위해 시계방향/반시계 방향을 모두 할 수 있도록 각각 메뉴를 제공하는 것도 하나의 해결책이 될 수 있다.

3. 바둑판 회전/대칭 이동에 대한 직관적 해결책

앞 장에서 언급한 것처럼 $y=x$ 에 대한 대칭 이동, $y=-x$ 에 대한 대칭 이동, 반시계 방향 회전을 새롭게 메뉴에 추가하는 것은 기존 인터페이스를 확장하는 개념이다.

그러나 이런 방식의 인터페이스는 바둑판의 회전/대칭 이동을 위해 사용자로 하여금 구체적인 일련의 수학적 연산을 컴퓨터에게 지시하게 하는 것으로 사용자의 처리 부담이 큰 비직관적인 방식이다.

이 처리 단계를 구체화 시키면 사용자는 바둑판을 회전/대칭 이동시키기 위해 다음과 같은 절차를 밟아야 한다.

- ① 연산 대상(가령 우상귀 소목 흑돌 A)의 위치를 선정한다.
- ② 그 대상이 회전/대칭 연산에 의해 옮겨져야 할 목적지 위치(가령 흑돌 B)를 선정한다.
- ③ 연산 대상이 목적지 위치로 옮겨지기 위한 구체적인 수학적 연산을 사용자가 해당 연산 메뉴를 클릭하여 컴퓨터가 수행하게 한다.

위 절차에서 ①, ②는 컴퓨터가 대신해 줄 수 없는 것인 반면 ③은 사람이 수행해야 할 이유가 전혀 없다. 컴퓨터는 이러한 수학적 연산은 오류 없이 순식간에 처리할 수

있다.

그러므로 본 연구에서 제안하고자 하는 인터페이스는 ‘회전/대칭’ 메뉴를 선택한 후 직접 사용자가 ①에서의 위치와 ②에서의 위치를 바둑판 위에서 마우스로 클릭해 주면 컴퓨터가 자동으로 회전/대칭 연산을 수행하도록 ③의 절차를 컴퓨터 프로그램에 완전히 맡기는 것이다.

③의 절차는 표 2에 의해 쉽게 수학적으로 구현될 수 있다. 이 표에서 바둑돌이 좌표 (x, y) 에 놓여 있을 때 원점에 대해 Θ 만큼 회전한 좌표는 (x', y') 가 된다. 여기서 $\Theta = \text{각도} * \pi$ 이다.

표 2 좌표 변환 공식

연산	변환 공식
x 축 대칭	$y' = -y$
y 축 대칭	$x' = -x$
직선 $y=x$ 대칭	$x' = y$ $y' = x$
직선 $y=-x$ 대칭	$x' = -y$ $y' = -x$
Θ 회전	$x' = x\cos\Theta - y\sin\Theta$ $y' = x\sin\Theta + y\cos\Theta$
원점 대칭	$x' = -x$ $y' = -y$

180° 회전은 원점 대칭과 같은 효과로 Θ 회전 공식에 의한 방법, $y=-x$ 대칭 후 $y=x$ 대칭하는 방법, $x' = -x, y' = -y$ 에 의한 방법이 있다. 여기서 맨뒤의 방식이 속도면에서는 가장 빠르나 회전 공식의 적용에 대한 일관성을 위해서는 Θ 회전 공식이 선호된다.

4. 결론

컴퓨터가 쉽게 할 수 있는 일을 사람이 힘들여 가며 지속적으로 하도록 하는 프로그램은 마땅히 개선되어야 할 것이다. 이런 관점에서 본 연구는 바둑판을 회전/대칭 이동시키기 위한 인터페이스를 사용자에게 편리하도록 직관적인 방식을 제시하였다.

바둑을 두는 사람은 자신이 선호하는 제1수의 위치가 있고, 그 위치에서 기보가 전개될 때 그 기보에 대한 이해도가 증진된다. 또한 자신이 백돌을 쥐었을 때, 다양한 위치에 두어지는 흑돌 1수를 고려하여 기보를 감상하려고

할 때 바둑판을 회전/대칭 이동시켜 보는 것은 사용자에게 필수적으로 필요한 기능이다.

그러므로 바둑판을 회전/대칭 이동시키기 위해 본 연구에서 제안한 연산 대상 위치와 목적지 위치만을 지칭하도록 하는 WYSWYG 방식의 인터페이스는 바둑 애호가뿐만 아니라 프로 기사들에게도 매우 편리한 기능이 될 것임에 틀림없다.

참고문헌

- [1] Rui Jiang, MultiGo 4.2.1, <http://ruijiang.com/multigo/>
- [2] 김경아, 바둑에서 형세판단에 근거한 효율적인 정석 선택, 경북대학교 컴퓨터공학과 석사학위 논문, 1994.
- [3] 박현수, 컴퓨터 바둑에서 안정도와 String Graph를 이용한 형세 평가, 경북대학교 컴퓨터공학과 박사학위 논문, 2004.
- [4] 김영상, 유기영, 확률적 평가에 기반한 컴퓨터 바둑의 후보 생성 시스템, 대한전자공학회논문지CI, 37권 2호, pp.21-30, 2000.
- [5] 최영숙, 순서화된 AND/OR Pruning을 이용한 바둑 사활의 효율 개선, 경북대학교 컴퓨터공학과 석사학위 논문, 1992.
- [6] 이경자, 후보 선정 알고리즘을 이용한 바둑 끝내기에 관한 연구, 경북대학교 컴퓨터공학과 석사학위 논문, 1992.
- [7] 성재경, 김용국, PDA 기반 네트워크 바둑 게임의 유저 인터페이스 연구, 정보과학회 학술대회발표 논문집, pp.496-498, 2003.
- [8] 기균도, 바둑 기보 인식 및 복기 시스템의 설계 및 구현, 연세대학교 대학원, 1995.
- [9] 최일호, 김진환, 바둑 수 위치 표시 방법에 대한 연구(규기점 좌표 방식의 제안), 명지대학교 예체능 연구소 예체능 논문집, 14권, 2003.