X-ray Attacks (Bitboards) (/X-ray+Attacks+%28Bitboards%29)

Table of Contents

The term **X-ray attack** was apparently originated by Kenneth Harkness . On page 25 of the April 1947 Chess Review , in his series *Picture Guide to Chess*, he mentioned forks and then wrote [1]:

```
There is another type of double attack in which the targets are threatened in one direction. The attacking piece
threatens two units, one behind the other, on the same rank, file or diagonal. This double threat has lacked a good
descriptive name. We suggest 'X-Ray' attack.
```

Attacking through

There are several tactical motives, where a sliding piece attacks through any other piece in a ray-direction:

Motiv	Blocker	Target behind Blocker
Battery	own sliding piece	any square or piece
Discovered Attack	own piece may remove	any opposing piece
Discovered Check	own piece may remove	opposing king
Pin	opposing piece	opposing valuable piece
Absolute Pin	opposing piece	opposing king
Partial Pin	opposing sliding piece	opposing valuable piece
Skewer	opposing valuable piece	opposing piece
X-ray	opposing sliding piece	own piece

Single Sliders

The single sliding piece, a rook, bishop or queen is specified by square index, likely from a bitscan of a piece-wise bitboard serialization of a sliding piece bitboard from a standard board-definition.

Modifying Occupancy

Following routines may be used to get all kind of x-ray attacks or defenses through the blockers for one particular slider. The idea is to intersect the sliding attacks with the desired blockers, which are subset of occupied. In a second run, those blockers are removed from the occupancy, to get the x-ray attacks as the symmetric difference of both attacks.

```
U64 xrayRookAttacks(U64 occ, U64 blockers, enumSquare rookSq) {
   U64 attacks = rookAttacks(occ, rookSq);
   blockers &= attacks;
   return attacks ^ rookAttacks(occ ^ blockers, rookSq);
}

U64 xrayBishopAttacks(U64 occ, U64 blockers, enumSquare bishopSq) {
   U64 attacks = bishopAttacks(occ, bishopSq);
   blockers &= attacks;
   return attacks ^ bishopAttacks(occ ^ blockers, bishopSq);
}
```

Alternatively one may use a condition to save the second lookup - one may use disjoint line-attacks rather than piece-attacks. If so, it makes sense to exclude the outer squares from the blockers.

```
U64 xrayFileAttacks(U64 occ, U64 blockers, enumSquare rookSq) {
   U64 attacks = fileAttacks(occ, rookSq);
```

```
   blockers &= attacks & C64(0x00FFFFFFFFFFFF00);
   if ( blockers == 0 ) return blockers;
   return attacks ^ fileAttacks(occ ^ blockers, rookSq);
}

U64 xrayRankAttacks(U64 occ, U64 blockers, enumSquare rookSq) {
   U64 attacks = rankAttacks(occ, rookSq);
   blockers &= attacks & C64(0x7E7E7E7E7E7E7E7E);
   if ( blockers == 0 ) return blockers;
   return attacks ^ rankAttacks(occ ^ blockers, rookSq);
}

U64 xrayDiagonalAttacks(U64 occ, U64 blockers, enumSquare bishopSq) {
   U64 attacks = diagonalAttacks(occ, bishopSq);
   blockers &= attacks & C64(0x007E7E7E7E7E7E00);
   if ( blockers == 0 ) return blockers;
   return attacks ^ diagonalAttacks(occ ^ blockers, bishopSq);
}
```


X-rays with one Lookup

Another idea is to consider all kind of blockers in one run, and to traverse intersections of the x-rays attacks with different target sets, to determine the blocking piece inside a loop. In the context of absolute pins, Oliver Brausch introduced to lookup x-ray attacks through any blocker in his engine OliThink [2] - similar to any occupied lookup technique to get sliding piece attacks. Instead of the pre-calculated line-attacks we can also pre-calculate attacks behind the first blocker, including a possible second blocker. The outer squares are redundant as well. Some sample on a rank:

```
rook      00001000
occupied  01011010
attacks   00010110
x-ray     01100001
```

Since absolute pins and discovered checkers are usually quite rare, it might be worth to spend some additional memory, e.g. 8 KByte for kindergarten - x-rays. Also the switch approach is great to consider all kind of x-ray stuff on a line.

Multiple Sliders

If keeping eight disjoint ray-direction attacks from fill-routines like Kogge-Stone Algorithm one may intersect them with desired blockers - and to perform another fill with that intersection:

```
U64 xrayAttacks(U64 sliders, U64 empty, U64 blockers, int dir8) {
   U64 attacks = slidingAttacks (sliders, empty, dir8);
   blockers &= attacks; // & notOuterSquares[dir8];
   if ( blockers )
      return slidingAttacks (blockers, empty, dir8);
   return 0;
}
```

Blockers by Targets

The other idea is to fill the potential target set, and to intersect it with the opposite ray-direction fill of the sliders and the relevant blockers:

```
U64 betweenSliderAndTarget(U64 sliders, U64 empty, U64 targets, int dir8) {
   U64 attacks    = slidingAttacks (sliders, empty, dir8);
   U64 fromtarget = slidingAttacks (targets, empty, opposite(dir8));
   return attacks & fromtarget;
}
```

See also

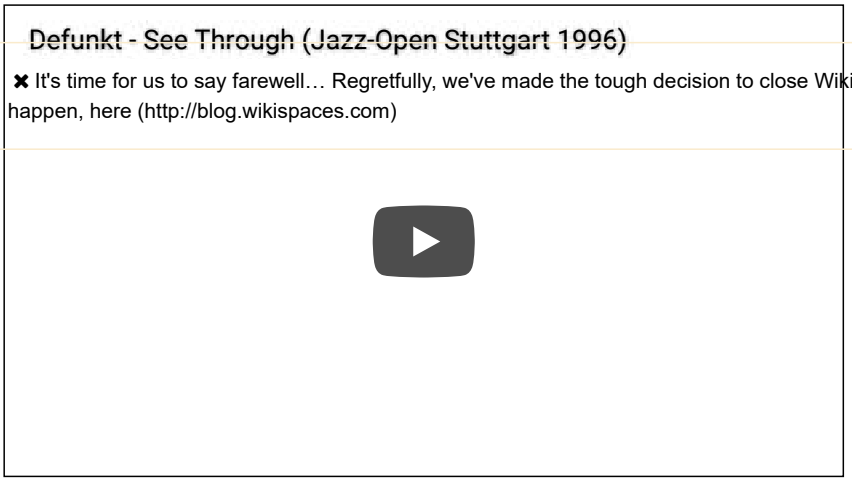
- Sliding Piece Attacks in OliThink

Forum Posts

- Generating "through" attacks with rotated bitboard by Vlad Stamate, CCC, August 28, 2009 » Rotated Bitboards

External Links

- X-ray (chess) from Wikipedia
- Defunkt - See Through, Jazzopen Stuttgart 1996, 3sat broadcast, YouTube Video
  lineup: Joseph Bowie , Kelvyn Bell , Larry Bowen , Bahnamous Bowie , Byron Bowie , Kim Clarke, Scooter Warner , Ronny Drayton



References

1. ^ Jack O'Keefe in Chess Note 4245. X-ray attack by Edward Winter
2. ^ Re: Question about SEE (Static exchange evaluation) by Oliver Brausch, CCC, December 18, 2007

What links here?

Page	Date Edited
Battery	Jan 29, 2017
Berkeley Chess Microprocessor	May 22, 2013
Bitboards	Nov 14, 2017
Checks and Pinned Pieces (Bitboards)	Aug 14, 2013
Discovered Attack	Nov 8, 2010
Discovered Check	Feb 1, 2018
Double Check	Apr 4, 2013
King Safety	Feb 14, 2018
Lyudmil Tsvetkov	Mar 28, 2018
Mate at a Glance	Sep 24, 2014
Morph	Dec 7, 2017
OliThink	May 19, 2017
Oliver Brausch	May 19, 2017
Pin	Mar 7, 2017
Rays	Oct 20, 2016
Rotated Bitboards	Mar 7, 2017
Rotated Indices	Oct 9, 2017
SEE - The Swap Algorithm	Jun 5, 2017
Skewer	Oct 22, 2014
Sliding Pieces	Aug 3, 2017
Tactics	Jan 12, 2018
Vlad Stamate	Dec 31, 2014
X-ray	Apr 12, 2017
X-ray Attacks (Bitboards)	Mar 31, 2015

Up one Level