

[Home](#) * [Board Representation](#) * [Bitboards](#) * [Pawn Pattern and Properties](#) * **Attack Spans**

Set-wise pattern, based on **attackspans** are for instance [isolated pawns](#), [passers](#) and [candidates](#).

The admittedly arbitrary definition of **front attackspans** includes the attacked squares itself, thus it is like a **fill** of [attacked squares](#) in the appropriate direction. The **rear attackspan** is the **complement** set on the **same file**, so that the union of both spans covers the **whole file**. As always, we keep disjoint east-west sets for separate applications, despite illustrated as union with one single pawn:

white attack frontspan	white attack rearspan	attack filefill with least one square attacked by white
. . 1 . 1 1 . 1 . . .
. . 1 . 1 1 . 1 . . .
. . 1 . 1 1 . 1 . . .
. . 1 . 1 1 . 1 . . .
. . . w 1 w 1 1 w 1 . . .
. 1 . 1 1 . 1 . . .
. 1 . 1 1 . 1 . . .
. 1 . 1 1 . 1 . . .
black attack frontspan	black attack rearspan	attack filefill with least one square attacked by black
. 1 . 1 1 . 1 . . .
. 1 . 1 1 . 1 . . .
. 1 . 1 1 . 1 . . .
. 1 . 1 1 . 1 . . .
. . . b 1 b 1 1 b 1 . . .
. . 1 . 1 1 . 1 . . .
. . 1 . 1 1 . 1 . . .
. . 1 . 1 1 . 1 . . .

The routines based on [one step only](#), [vertical fills](#) and [pawn spans](#) :

```
U64 wEastAttackFrontSpans (U64 wpawns) {return eastOne(
wFrontSpans(wpawns));}
U64 wWestAttackFrontSpans (U64 wpawns) {return westOne(
wFrontSpans(wpawns));}
U64 bEastAttackFrontSpans (U64 bpawns) {return eastOne(
bFrontSpans(bpawns));}
U64 bWestAttackFrontSpans (U64 bpawns) {return westOne(
bFrontSpans(bpawns));}

U64 wEastAttackRearSpans (U64 wpawns) {return eastOne(
wRearFill(wpawns));}
```

```
U64 wWestAttackRearSpans (U64 wpawns) {return westOne(
wRearFill(wpawns));}
U64 bEastAttackRearSpans (U64 bpawns) {return eastOne(
bRearFill(bpawns));}
U64 bWestAttackRearSpans (U64 bpawns) {return westOne(
bRearFill(bpawns));}
```

The filefills are independent on color

```
U64 eastAttackFileFill (U64 pawns) {return eastOne(fileFill(pawns));}
U64 westAttackFileFill (U64 pawns) {return westOne(fileFill(pawns));}
```

- If a pawn is not member of own west or east attack filefills, the pawn is [half-isolated](#).
- If a pawn is not member of the union of own east and west attack filefills, the pawn is [isolated](#).
- If a pawn is not member of the union of all opponent front pawn- and attack-spans, it is a [passed pawn](#).

What links here?

Page	Date Edited
Attack Spans	Mar 4, 2010
Backward Pawns (Bitboards)	May 5, 2017
Candidates (Bitboards)	Nov 11, 2017
Fill Algorithms	Nov 6, 2013
Holes	Oct 16, 2017
Isolated Pawns (Bitboards)	Nov 18, 2012
Passed Pawns (Bitboards)	Aug 23, 2010
Pawn Pattern and Properties	Nov 11, 2017
Pawn Spans	Jul 22, 2010
PawnKing	Sep 5, 2014
Pin	Mar 7, 2017

[Up one Level](#)