

Quantifying evaluation features

by Mark Watkins

1 Introduction and methodology

April 18, 2011

The purpose of this document is to provide a quantification of the overlap of evaluation features of various chess engines, with particular reference to the case of Fruit 2.1 and various Rybka versions.

1.1 Evaluation features and originality

One initial abstract question to answer is whether a large overlap of evaluation features should be considered a breach of originality. I might point out that most chess engine authors experiment with and accept/reject a large number of features, so that a wholesale copying of someone’s specific list (with minor additions and/or changes over a period of development) would likely represent a large competitive advantage in the engine development process. Much more typical would be to pick-and-choose various features from a broad spectrum of engines and other sources, and see which combinations work best for oneself.¹

1.1.1 Levels of abstraction (analogy to copyright)

Although the ICGA investigation is not centered on copyright, there are many philosophical considerations that inform both copyright law and the concept of originality as pertinent to the ICGA. In this genre, Jeremy Bernstein provided a PDF of a 1988 *Boston College Law Review* paper by J. Dianne Brinson entitled *Copyrighted Software: Separating the Protected Expression from Unprotected Ideas, A Starting Point*. The author (§II-D, 811–15) discusses “substantial similarity”, and after pointing out that copyright does not protect abstract ideas, recalls the example in the standard treatise *Nimmer on Copyright* (1987), wherein the two plays *Romeo and Juliet* and *West Side Story* were compared to illustrate the idea/approach distinction “identifying thirteen common elements and concluding that the common elements form a pattern sufficiently concrete to provide a basis for a finding that the two works are substantially similar.” The same could be said of (say) a movie adaptation of a book, as the “plot” remains more-or-less the same, while many individual elements are changed (either by necessity or volition) so as to fit the new communication medium.²

Brinson also discusses “levels of abstraction”, with generic concepts (*e.g.*, a program that plays chess) at one end and specific implementations at the other. A large overlap in evaluation features seems to come fairly near the “specific implementations” part of this spectrum, though indeed, part of the purpose of this document is to try to quantify this in the given case.

¹An imperfect analogy could be the following. If a teacher asked a student to write a 10-page report on Argentina, and received essentially the same material as in *Encyclopedia Britannica* (though reworded), this could hardly be considered original. This would be true even if said student happened to be a dance aficionado, and thus wrote enthusiastically in his/her own words in an elongated section on the tango. Conversely, if the student had taken five or so sources and formed a composite from them, any specific parts of the underlying content would be not be novel by any means, but the overall presentation would be original.

²From the standpoint of copyright, the fact that the re-use of evaluation features was again in a chess engine (rather than say, chess training software) is also of import.

1.1.2 Evaluation features versus numerology

Even upon assuming that evaluation features can be placed into a framework of abstraction, it still remains but one component of originality, and the weight to be given to it is not immediately clear. For instance, it could be argued that numerology attached to such features is more important than the features themselves (here I speak in terms of originality rather than merely engine strength). This would certainly be true if the features were broadly stated at an abstract level, as noted by Hayes/Levy in their 1974 book on the first World Chess Computer Championship (see page 62, in the section *Numerical Evaluation of Positions*). However, when specific conditions regarding the features recur, there is at the very least some impact on the concept of originality, though it seems to me that opinion of others is needed regarding where this sits in the overall scheme of things. It must be stressed that the analysis here only rarely diverges into consideration of *how* the specifics of an evaluation feature are implemented (for instance, considering questions of chessboard representation), but rather focusses on *what* the programs actually compute.

A final aspect here might be one of temporality, in that data-mining for new features might be in vogue at one time, while tuning the numerology could take precedence at another. Due to this it can become less clear on what basis to decide that something is “original” as opposed to an “automatic” process.

1.1.3 Whether evaluation features can and should be measured

Another question is whether evaluation features can actually be measured with any validity. As Brinson notes (citation as above), in the context of copyright, unprotected ideas have been separated from protected expression for such widespread areas as ‘fiction and plays, non-fiction, songs, labels, jewelry, games and contests, works of visual art, television show characters, and videogames.’ While the methods used here would not meet the most stringent notion of “scientific”, it seems there is adequate precedent for making more “quasi-scientific” conclusions that are reliant on means that are not necessarily completely objective.

1.2 Methodology

A number of chess engines (most of them open-source) were taken and their evaluation features examined. For every “feature” there was a score given from 0.0 to 1.0 on whether there was “overlap” with every other engine. The score of 1.0 was given when a feature was more-or-less *exactly* the same (though of course numerology is a different issue, in general), while 0.0 would be that an engine doesn’t have a feature. The typical score given to a partial match was 0.5, though this would vary depending on how close the features were deemed to be (this is the most subjective part of the process). Typically, for every matching sub-condition of a feature another tenth of a point would be added, with a subtracting if they had competing sub-conditions. If there was only a minor or almost tangential difference, a score of 0.8 would be given. Some effects of the use of “partial credit” (and more generally the allocation of features) are considered more below.

The scores for all features were then summed. For each pair of engines, this gave two scores, X/Y and X/Z where X is the common overlap, and Y and Z

are the total count of features for the engines. It is not exactly clear (to me) what the best method for determining “total” overlap, and so for now these are just added to get $2X/(Y + Z)$, and a percentage score taken from that.³ The comparisons other than between Fruit 2.1 and Rybka versions (and of course Rybka-Rybka comparisons) formed a “control” group, from which a “typical” overlap in evaluation features could be determined.

I decided not to include PST as a feature, as differences there would often be more numerical than methodological (though some engines do lack PST for certain pieces, or have an additional king PST for the endgame).

1.2.1 Choice of engines

Currently there are 9 engines in the comparison: Rybka 1.0 Beta, Rybka 2.3.2a, and Fruit 2.1 of course, and various pre-Fruit engines like Crafty 19.0, RESP 0.19, Phalanx XXII, Pepito 1.59, Faile 1.4, and EXchess 5.01beta. The evaluation function of Rybka 1.0 Beta is almost identical to that in Rybka 2.3 (Feb 2007), with only three or four modifications of features (drawishness from pawn files added, lazy eval criterion changed, interpolation became linear, and tempo was removed), and the same numerology kept throughout (except for some material imbalance perturbations). There was some debate on whether to include either “Rybka 1.6.1” (late 2004) or Rybka 3, but it was decided not to so because these were respectively private and non-free.

It may be beneficial to add engines that at least partially post-dated Fruit 2.1, such as Glaurung, Scorpio, GreKo, or Cyrano, but I wait for guidance on which such engines to use.

1.2.2 Result

The final numbers for “evaluation feature overlap” can be seen in Table 49. After removing one of the Rybka versions and excluding the Fruit/Rybka data point, the mean from the other 27 engine pairings is about 31.3%, with the standard deviation a bit over 5.6%. As a first approximation (which is not quite correct), if we took the distribution to be Gaussian and the data here to be representative, the overlap between Fruit/Rybka1 overlap at 74.4% would exceed 7.5σ , while that of Fruit/Rybka232 would be nearly 6σ with less than 1 in 10^8 chance of this happening “randomly” from a suitable data set.⁴ The next largest deviation in the data is the RESP 0.19 overlap with EXchess 5.01beta, which is about 2.2σ , slightly large for the sample size, but not abnormally so.

At the very least, this result should shift the burden to be on anyone who would find the Rybka/Fruit overlap non-extraordinary to produce an example of “accepted” engines which possess common features of a similar magnitude. I expect that even evaluation functions written by the same person, such as Tord Romstad with Glaurung and Gothmog, or Larry Kaufman with Rybka 3 and Komodo,⁵ would have significantly smaller overlap than those for Fruit 2.1 and the Rybka versions investigated here.

³One case where this could become a problem is when X/Y is near 1.0 but X/Z is not, particularly if the Z -engine was temporally first.

⁴I personally do not place much value on the exact numbers, but this at least gives a general indication of the likelihood of the Fruit 2.1 match of evaluation features with the Rybkas.

⁵Though it hardly needs mentioning, it is of course the case that the cross-over in authorship for both of these involves no clandestine elements.

1.2.3 Robustness to small changes

The conclusive nature of the above result makes it quite robust to any small disputes about specific numbers or methodological choices. A more general question might be how the evaluation features here were chosen. Some of them are quite precise (like mobility split into knight mobility, bishop mobility, *et cetera*), while others are not quite so delineated, the most notable being how to judge king safety and passed pawns. In particular, I didn't describe each king safety element individually piecewise, as most engines treat all pieces more-or-less the same once a condition is chosen; contrariwise, mobility often does vary in its specifics from piece to piece. It might be useful to have an additional "relative mobility scaling" feature as with the relative king safety scaling amongst pieces, but I chose not to do so. In general, a feature is the "smallest" possible that makes sense in a chessic sense. An example might be open files, where there can be many modifiers in a specific implementation, but breaking these down into subfeatures is likely to dwell too much on minutiae. As indicated at the top of this paragraph, it seems quite unlikely to me that anything but a drastic change of method would lead to a different result.

There is also the question of the size of "partial credit" to give. For instance, Fruit 2.1 and the Rybkas have a lot of direct overlaps that receive 1.0 scores, and so a general lifting of partial credit scores would tend to diminish the strikingness of their overlap. However, it seems to me that a 1.0 overlap should be given a higher priority, as many of the side conditions are nontrivial and in many cases are not likely to engender an exact match merely by accident. Another facet of the final combination of scores is that an 0.2 overlap will count 0.4/2.0, which might have more impact than 0.0/1.0 for some engine pairings.

1.2.4 Other Rybka versions

As noted above, there are hardly any evaluation changes between Rybka 1.0 Beta (Dec 2005) and Rybka 2.3 (Feb 2007). In fact, the overlap with Fruit 2.1 is *larger* for Rybka 2.3 than Rybka 1.0 Beta; the change of lazy evaluation is irrelevant as Fruit 2.1 lacks it, the addition of drawishness from pawn structures is offset by the removal of the tempo bonus (again neither one being in Fruit 2.1), and so the fact the interpolation became linear implies that the overlap is slightly larger.

2 Enumeration of features

2.1 Knights

2.1.1 Knight mobility

Knight mobility is a bonus for the number of pseudolegal moves (or alternatively squares attacked) by a knight. Variations can include whether the squares are in front of the knight, whether the squares are guarded by an enemy piece/pawn, etc. Many engines omit this feature. As an example, in his 1989 description of ANALYSE, Hartmann noted that he found no specific value to it.

Both Fruit 2.1 and Rybka 1.0 Beta give a bonus linear in the number of pseudolegal moves. However, Fruit 2.1 starts the mobility count at -4 , while Rybka 1.0 Beta starts it at zero. Pepito 1.59 has a `MOV_CABALLO` array with nonlinear adjustments, but it is commented out in this specific version, and squares attacked by the enemy pawns were ignored. The remaining engines have no knight mobility bonus.

Table 1: Knight mobility

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	1.0	0.0	0.0	0.0	0.8	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.8	0.0	0.0	0.0	1.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.1.2 Knight outposts

This feature gives a bonus to a knight that occupies a position where it cannot easily be driven off. Variations include whether the knight is guarded, and whether the opponent can easily trade off the knight.

Crafty 19.0 first determines whether the square the knight occupies is valuable enough to be considered an outpost (via an array), and then determines if there are any enemy pawns that could possibly attack that square. A square-based bonus is added at this point, and if we guard the outpost square with a pawn ourselves, the bonus is increased, either by 50% or doubling when two pawns guard the knight or the opposing side has no applicable minors (checking bishop color) to trade for it. Phalanx XXII first checks that either the knight is right in front of an enemy pawn, or that it is guarded by a pawn and is on a “good” square (from an array). If the opponent further cannot attack the knight with a pawn, a bonus is given depending on the previous aspects (whether an enemy pawn is blocked, whether the square is “good”, and whether the knight is guarded) with the bonus varying depending on the square in the first two cases. An additional bonus (depending on the square) is added if the opponent has no applicable minor to trade off.

Pepito 1.59 first gives a square-based bonus if no enemy pawn can attack the knight, and then adds 50% more when the knight is guarded by a pawn, and then 200% more when the knight is doubly guarded by pawns. EXchess 5.01beta computes outpost squares in the pawn eval, and seems to require it to be guarded by a friendly pawn, and not attackable by an enemy pawn. A square-based bonus is then applied when an outpost square is occupied by a knight. RESP 0.19, Faile 1.4, Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a lack knight outposts.

Table 2: Knight outposts

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.0	0.0	0.5	0.0	0.0	0.7	0.4
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.5	0.0	0.0	0.0	1.0	0.0	0.0	0.4	0.4
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.7	0.0	0.0	0.0	0.4	0.0	0.0	1.0	0.6
EX5b	0.4	0.0	0.0	0.0	0.4	0.0	0.0	0.6	1.0

2.1.3 Knight in opposing corner or trapped

This feature gives a penalty to a knight that is trapped, or more generally in the opposing corner (**a8/h8** for White). The penalty can either be PST-based, or use SEE and attacked squares to determine whether the knight is really trapped.

Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a all give a PST penalty of about 100 centipawns to a White knight on **a8/h8**. Phalanx XXII has code for **wNa8** that checks for **bPa7** and that the knight cannot safely escape via **c7** (the latter via SEE). It also has code for **wNa7/bPb7** with either **bPc6/a6**. The penalty is 1.4 pawns in the former case, and 0.45 in the latter. Pepito 1.59 also checks for knights at **a7/a8**, with the same condition as Phalanx XXII for **a7**, and with a quite similar one for **a8** (there is a **popcnt** of attackers/defenders of **c7**, rather than SEE). The penalty is the same in both cases (150cp). Crafty 19.0, Faile 1.4, EXchess 5.01beta, and RESP 0.19 seem to lack this feature.

Table 3: Knight in opposing corner or trapped

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	1.0	1.0	0.4	0.0	1.0	0.4	0.0
R232	0.0	0.0	1.0	1.0	0.4	0.0	1.0	0.4	0.0
Phal	0.0	0.0	0.4	0.4	1.0	0.0	0.4	0.7	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	1.0	1.0	0.4	0.0	1.0	0.4	0.0
Pepi	0.0	0.0	0.4	0.4	0.7	0.0	0.4	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.2 Bishops

2.2.1 Bishop mobility

Bishop mobility is a bonus for the number of pseudolegal moves (or alternatively squares attacked) by a bishop. Variations can include whether the squares are in front of the bishop, whether the squares are guarded by an enemy piece/pawn, etc. Many engines use some sort of variation with a pawn skeleton and immobile pawns, partially covering the case of bad bishops.

The following engines give a bonus linear in the number of pseudolegal moves: Fruit 2.1, Rybka 1.0 Beta, Rybka 2.3.2a, Crafty 19.0, RESP 0.19. However, Fruit 2.1 starts the mobility count at -6 , while the rest start it at zero. Next, EXChess 5.01beta has a linear bonus, but the accounting of moves is modified by pawn considerations. Pepito 1.59 counts pseudolegal moves by the bishop, excluding moves to squares attacked by enemy pawns. The bonus is then non-linear in this count, penalising low mobility rather drastically. Phalanx XXII also has a non-linear bonus of a similar style, but instead counts squares attacked, ignoring friendly pawns that are immobile, and double-counting enemy pieces. Faile 1.4 lacks this feature.

Table 4: Bishop mobility

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	1.0	1.0	1.0	0.3	0.0	0.8	0.5	0.7
RESP	1.0	1.0	1.0	1.0	0.3	0.0	0.8	0.5	0.7
Ryb1	1.0	1.0	1.0	1.0	0.3	0.0	0.8	0.5	0.7
R232	1.0	1.0	1.0	1.0	0.3	0.0	0.8	0.5	0.7
Phal	0.3	0.3	0.3	0.3	1.0	0.0	0.3	0.4	0.4
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.8	0.8	0.8	0.8	0.3	0.0	1.0	0.5	0.6
Pepi	0.5	0.5	0.5	0.5	0.4	0.0	0.5	1.0	0.4
EX5b	0.7	0.7	0.7	0.7	0.4	0.0	0.6	0.4	1.0

2.2.2 Trapped bishops

Trapped bishops is a penalty given to bishops that are trapped by an enemy pawn configuration. There are various differences in the chosen patterns here.

Fruit 2.1 gives a penalty for `wBa7/pPb6` or `wBb8/pPc7`, and a penalty half as large for `wBa6/pPb5`. Rybka 1.0 Beta and Rybka 2.3.2a penalise all three of these configurations uniformly, with the very minor difference that Rybka 1.0 Beta allows at most one trapped bishop per side, while Rybka 2.3.2a allows multiple. Crafty 19.0 penalises `wBa7/pPb6` only (in comments it says that a pawn that can move to `b6` suffices, and that the pawn must be defended, but the code differs in both aspects), as does RESP 0.19. Next, Phalanx XXII only penalises `wBa7/pPb6`, and has a SEE check on the pawn. Pepito does almost the same, requiring that Black guards the pawn. EXChess 5.01beta also only has `wBa7/pPb6/pPc7`, with both pawns required. Faile 1.4 lacks this feature.

Table 5: Trapped bishops

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	1.0	0.5	0.5	0.7	0.0	0.5	0.5	0.4
RESP	1.0	1.0	0.5	0.5	0.7	0.0	0.5	0.5	0.4
Ryb1	0.5	0.5	1.0	0.9	0.3	0.0	0.7	0.3	0.3
R232	0.5	0.5	0.9	1.0	0.3	0.0	0.8	0.3	0.3
Phal	0.7	0.7	0.3	0.3	1.0	0.0	0.3	0.8	0.7
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.5	0.5	0.7	0.8	0.3	0.0	1.0	0.3	0.3
Pepi	0.5	0.5	0.3	0.3	0.8	0.0	0.3	1.0	0.8
EX5b	0.4	0.4	0.3	0.3	0.7	0.0	0.3	0.8	1.0

2.2.3 Opposite Bishop ending

This feature corresponds to a reduction in score for an ending involving opposite-coloured bishops. There are many possible variations, such as only applying it to strict bishop endings (B vs B of opposite colours with pawns), or allowing it more generally. The score adjustment also admits variations, such as only reducing the score when the pawn imbalance is small, or reducing only the material part of the score.

Fruit 2.1 and Rybka 1.0 Beta both reduce the score by a factor of 2 in strict opposite bishop endings when the pawn imbalance is 2 or less. Rybka 2.3.2a does the same, except that (due to bit-packing) the score is also halved when the pawn imbalance is 6 or more. For Crafty 19.0, if each side's piece-material is 11 or less and there are opposite bishops, then if the material imbalance exceeds 2, the score is halved in a strict bishop ending, and otherwise halves the non-material part; if the material imbalance is 2 or less, then the score is divided by 4 in a strict opposite bishop ending. Phalanx XXII applies a reduction only in strict bishop endings, first halving the score and then adding back 50 centipawns and 25% of any material difference. Pepito 1.59 sets a `final_tablero` variable in strict opposite bishop endings, and halves the score. RESP 0.19, EXChess 5.01beta, and Faile 1.4 lack this feature.

Table 6: Opposite Bishop ending

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.5	0.5	0.5	0.0	0.5	0.4	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.5	0.0	1.0	0.9	0.6	0.0	1.0	0.7	0.0
R232	0.5	0.0	0.9	1.0	0.6	0.0	0.9	0.7	0.0
Phal	0.5	0.0	0.6	0.6	1.0	0.0	0.6	0.7	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.5	0.0	1.0	0.9	0.6	0.0	1.0	0.7	0.0
Pepi	0.4	0.0	0.7	0.7	0.7	0.0	0.7	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.2.4 Bishop outposts

This feature gives a bonus to a bishop that occupies a position where it cannot easily be driven off. Variations include whether the bishop is guarded, and whether the opponent can easily trade off the bishop.

Phalanx XXII treats bishop outposts much the same as knight outposts, with there being an additional constant bonus if there is an enemy pawn two squares in front of the bishop, and any bonuses being halved in the endgame. There is no condition concerning whether the bishop can be traded off. Pepito 1.59 gives a square-based bonus to a bishop that is guarded and cannot be attacked by an enemy pawn. EXchess 5.01beta treats bishop outposts exactly like knight outposts: upon computing the possible outposts from the pawn eval, it gives a square-based bonus to a bishop that occupies such a square. RESP 0.19, Faile 1.4, Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a lack bishop outposts. Crafty 19.0 has some condition about this, but it refers to blocking central pawns (and it possibly erroneous).

Table 7: Bishop outposts

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.3	0.4
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	0.3	0.0	0.0	1.0	1.0
EX5b	0.0	0.0	0.0	0.0	0.4	0.0	0.0	1.0	1.0

2.2.5 Bad bishops

This is a penalty that applies to bishops for which there are many friendly pawns on the same color. A variation could include whether such pawns are immobile. There is also a question of whether the penalty is applied generically, or only when a side has one bishop.

Crafty 19.0 penalises a bishop without partner by an amount linear in the number of friendly pawns on that color. Phalanx XXII penalises any White bishop that is the same colour as a blocked pawn on **d3/e3/e4/d4**, with the penalty doubled if the blocker is an enemy pawn. Pepito applies a penalty to a solitary bishop (no other pieces – this might be a bug) based upon the number of pawns on the color, using a table lookup (non-linear). RESP 0.19, Faile 1.4, Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a lack this feature. EXchess 5.01beta has a comment saying “good/bad bishops” and a `BAD_BISHOP` variable, but it is unused.

Table 8: Bad bishops

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.0	0.0	0.3	0.0	0.0	0.4	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.3	0.0	0.0	0.0	1.0	0.0	0.0	0.3	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.4	0.0	0.0	0.0	0.3	0.0	0.0	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.2.6 Bishop pair

This is a specific bonus for a bishop pair. Other features with material imbalances are considered elsewhere. Variations include questions of pawn count.

Fruit 2.1 has a constant bonus for the bishop pair. Rybka 1.0 Beta and Rybka 2.3.2a give a bonus in their material tables. It seems rather difficult to sort out exactly what type of bonus is applied, though it doesn't seem to depend on the number of pawns as much as what other pieces are left. Crafty 19.0 gives a bonus depending on the number of own pawns, the array being constant until 7 or 8 pawns is reached. Phalanx XXII gives a bonus partially dependent on the minimum mobility among the bishops, doubling any bonus in the endgame. Pepito 1.59 has a constant `BONUS_2_ALFILES` that is commented out, with it replaced by an array depending on the total number of pawns (the array has larger variation than Crafty's). RESP 0.19 gives a constant bonus for having a bishop pair, as does EXchess 5.01beta. Finally, Faile 1.4 seems to lack this feature.

Table 9: Bishop pair

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.7	0.3	0.3	0.3	0.0	0.7	0.5	0.7
RESP	0.7	1.0	0.3	0.3	0.3	0.0	1.0	0.5	1.0
Ryb1	0.3	0.3	1.0	1.0	0.3	0.0	0.3	0.3	0.3
R232	0.3	0.3	1.0	1.0	0.3	0.0	0.3	0.3	0.3
Phal	0.3	0.3	0.3	0.3	1.0	0.0	0.3	0.3	0.3
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.7	1.0	0.3	0.3	0.3	0.0	1.0	0.5	1.0
Pepi	0.5	0.5	0.3	0.3	0.3	0.0	0.5	1.0	0.5
EX5b	0.7	1.0	0.3	0.3	0.3	0.0	1.0	0.5	1.0

2.3 Rooks

2.3.1 Rook mobility

The rook mobility feature measures the number of pseudolegal moves (or alternatively squares attacked) by a rook. A related concept is (half)-open files, but that is handled separately (those are more related to “potential” mobility, it seems).

Fruit 2.1, Rybka 1.0 Beta and Rybka 2.3.2a all give a bonus linear in the number of pseudolegal moves. However, Fruit 2.1 starts the mobility count at -7 , while the Rybkas start it at zero. RESP 0.19 has a linear bonus, and also a penalty for a rook with 0 or 1 pseudolegal moves. Phalanx XXII gives a bonus according to an array, with 0 or 1 legal moves incurring a large relative penalty. There is also some component with forward mobility at a different point in the code. Crafty 19.0 has a penalty if a rook cannot move horizontally. Pepito 1.59 has a MOV_TORRE array that is unused. EXChess 5.01beta and Faile 1.4 also lack this feature.

Table 10: Rook mobility

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.5	0.2	0.2	0.3	0.0	0.2	0.0	0.0
RESP	0.5	1.0	0.7	0.7	0.4	0.0	0.6	0.0	0.0
Ryb1	0.2	0.7	1.0	1.0	0.4	0.0	0.8	0.0	0.0
R232	0.2	0.7	1.0	1.0	0.4	0.0	0.8	0.0	0.0
Phal	0.3	0.4	0.4	0.4	1.0	0.0	0.4	0.0	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.2	0.6	0.8	0.8	0.4	0.0	1.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.3.2 Rooks on open files

This feature counts whether a rook is on an open file. Variations can include whether the rooks are doubled on the file, whether the opponent has an immovable minor impeding the file, and whether the opposing king is on the file (this can be given either a direct bonus, or merged into king safety). The definition of an open file can also vary, such as whether `wRa3/wPa2` is open.

Fruit 2.1 first gives a general deduction of `Bonus/2`, and then gives a bonus for rook on a file with no pawns (interweaving this with semi-open files). Under a given material condition (queen and piece), if the enemy king is on the same or an adjacent file a bonus is given, with then a larger bonus if the king is on the same file. Rybka 1.0 Beta gives a bonus for an open file when there are no pawns in front of the rook. Under a given material condition (queen and piece), if the enemy king is on the same or an adjacent file a bonus is given, with then a larger bonus if the king is on the same file (and in front of the rook). Rybka 2.3.2a gives a bonus for an open file when there are no pawns in front of the rook. If the enemy king is on this file (and in front of the rook) an additional bonus is given. Crafty 19.0 gives a bonus for a rook on a file with no pawns, and increases a tropism count based on the file distance to the enemy king. There is also a penalty (of `Bonus/2`) when an open file exists and the rook cannot move to it (in one move). The comments talk about a friendly rook, but this is not implemented. Pepito 1.59 gives a bonus for a rook on a file with no pawns, and increases this when the rooks are doubled. There is `#if TROPISMO` code that is applicable to a file-based distance to the enemy king. Phalanx XXII deducts a constant penalty in the middlegame if the file of a rook has a friendly pawn; the same penalty also applies to an opposing pawn on the file, so that the overall effect for open files is a constant non-deduction.

Failé 1.4 gives a constant bonus for a rook on a file with no pawns. RESP 0.19 does the same, and additionally gives a penalty to a king that is on an open file when the opponent has a rook/queen (the location of the major is irrelevant) – it is not clear whether it is better to put this additional factor here, or in with king safety. EXChess 5.01beta gives a bonus for a rook on a file with no pawns, and an additional one if the rooks are doubled. It also has a `FILE_BONUS` variable that is unused.

Table 11: Rooks on open files

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.6	0.4	0.3	0.6	0.6	0.6	0.7	0.6
RESP	0.6	1.0	0.3	0.4	0.8	0.9	0.5	0.7	0.7
Ryb1	0.4	0.3	1.0	0.6	0.3	0.3	0.7	0.4	0.3
R232	0.3	0.4	0.6	1.0	0.4	0.4	0.3	0.4	0.3
Phal	0.6	0.8	0.3	0.4	1.0	0.9	0.5	0.7	0.7
Fail	0.6	0.9	0.3	0.4	0.9	1.0	0.5	0.7	0.7
Fr21	0.6	0.5	0.7	0.3	0.5	0.5	1.0	0.4	0.3
Pepi	0.7	0.7	0.4	0.4	0.7	0.7	0.4	1.0	0.8
EX5b	0.6	0.7	0.3	0.3	0.7	0.7	0.3	0.8	1.0

2.3.3 Rooks on semi-open files

This feature counts whether a rook is on a semi-open file. Variations can include whether the rooks are doubled on the file, whether the opponent has a weak pawn on the file, and whether the opposing king is on the file (this can be given either a direct bonus, or merged into king safety). The definition of a semi-open file can also vary, such as whether `wRa3/wPa2/bPa7` is semi-open.

Fruit 2.1 first gives a general deduction of `OpenFileBonus/2`, and then gives a bonus for rook on a file with no friendly pawns. Under a given material condition (queen and piece), if the enemy king is on an adjacent file a bonus is given, with a larger bonus if the king is on the same file. Rybka 1.0 Beta gives a bonus for a semi-open file when there are no friendly pawns in front of the rook. Under a given material condition (queen and piece), if the enemy king is on an adjacent file a bonus is given, with a larger bonus if the king is on the same file (and in front of the rook). Rybka 2.3.2a gives a bonus for a semi-open file when there are no pawns in front of the rook. If the enemy king is on this file (and in front of the rook) an additional bonus is given. Crafty 19.0 gives a bonus for a rook on a file with no friendly pawns, and increases a tropism count based on the file distance to the enemy king. Pepito 1.59 gives a bonus for a rook on a file with no friendly pawns, and increases this when the opponent has a weak pawn on the file. There is `#if TROPISMO` code that applies to a file-based distance to the enemy king.

Phalanx XXII gives a penalty to an isolated pawn on a semi-open file when attacked by a rook. It also deducts a constant penalty in the middlegame if the file of a rook has a friendly pawn; the same penalty also applies to an opposing pawn on the file. Faile 1.4 gives a constant bonus to rook on a file with no friendly pawns. RESP 0.19 does the same, and additionally gives a penalty to a king that is on a semi-open file when the opponent has a rook/queen (the location of the major is irrelevant) – it is not clear whether it is better to put this additional factor here, or in with king safety. EXChess 5.01beta gives a bonus for a rook on a half-open file with no friendly pawns, and an additional one if the rooks are doubled; the definition of “half-open” requires the enemy to have a weak pawn on it. It additionally has a `HALF_FILE_BONUS` variable that is unused.

Table 12: Rooks on semi-open files

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.6	0.4	0.3	0.3	0.6	0.6	0.7	0.4
RESP	0.6	1.0	0.3	0.4	0.3	0.9	0.5	0.7	0.5
Ryb1	0.4	0.3	1.0	0.6	0.3	0.3	0.6	0.4	0.2
R232	0.3	0.4	0.6	1.0	0.3	0.4	0.3	0.4	0.2
Phal	0.3	0.3	0.3	0.3	1.0	0.3	0.3	0.4	0.4
Fail	0.6	0.9	0.3	0.4	0.3	1.0	0.5	0.7	0.6
Fr21	0.6	0.5	0.6	0.3	0.3	0.5	1.0	0.4	0.3
Pepi	0.7	0.7	0.4	0.4	0.4	0.7	0.4	1.0	0.7
EX5b	0.4	0.5	0.2	0.2	0.4	0.6	0.3	0.7	1.0

2.3.4 Blocked rooks

This feature concerns when a king has moved so as to block a rook on its original square. Variations include the squares for the king/rook, and whether there are pawns in front of the rook.

Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a all penalise a White rook on a1/a2/b1 when the king is on b1/c1 (and symmetrically). Crafty 19.0 penalises wKh1/Rh2, wKg1/Rh1/Rh2, and wKf1/Rg1/Rh1/Rh2 (again with symmetries). Pepito 1.59 has a TORRE_ATRAPADA variable, but it is unused. Phalanx XXII uses the same pattern as Fruit/Rybka, but requires that a pawn be in front of the rook. EXChess 5.01beta penalises a BOXED_IN_ROOK in the opening for wKf1/Rg1/Rh1/Rg2/Rh2 and wKg1/Rh1/Rh2. Faile 1.4 and RESP 0.19 lack this feature.

Table 13: Blocked rooks

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.6	0.6	0.4	0.0	0.6	0.0	0.6
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.6	0.0	1.0	1.0	0.7	0.0	1.0	0.0	0.7
R232	0.6	0.0	1.0	1.0	0.7	0.0	1.0	0.0	0.7
Phal	0.4	0.0	0.7	0.7	1.0	0.0	0.7	0.0	0.4
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.6	0.0	1.0	1.0	0.7	0.0	1.0	0.0	0.7
Pepi	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EX5b	0.6	0.0	0.7	0.7	0.4	0.0	0.7	0.0	1.0

2.3.5 Rooks on 7th rank

This feature concerns bonuses for a rook on the 7th rank. This can be done in PST, or directly under certain conditions. Bonuses for secondary majors on the 7th can also apply (though the queen on the 7th bonus is handled separately).

Fruit 2.1 and Rybka 1.0 Beta both give a bonus to a rook on the 7th rank when the enemy king is on the 8th rank, or there is an enemy pawn on the 7th. Rybka 2.3.2a gives a bonus when there is a king or pawn on the 7th or 8th. Crafty 19.0 gives a bonus as with Fruit/Rybka1, and gives an additional bonus if the rook is guarded by another major on the 7th, and another when the Black king is on the 8th rank, there are no Black pieces on b7-g7, and White has a passed pawn. Pepito 1.59 gives a bonus for a rook on the 7th/8th when the opposing king is on the 8th rank, with the bonus depending on the number of majors on such ranks. Phalanx XXII counts 2 for rooks on 7th and 1 for queens on 6th/8th and determines a score from this, with an additional bonus if the opposing king is on the 8th rank. It also has a larger PST for 7th/8th ranks. Faile 1.4 gives a 7th rank bonus to a rook, slightly larger in the endgame. RESP 0.19 gives a bonus when two rooks are on the 7th, and (in queen evaluation) a bonus for queen and rook on 7th. The PST score gives a standard bonus (34 in opening, 15 in endgame). EXChess 5.01beta seems only to give a bonus via PST (and rather small).

Table 14: Rooks on 7th rank

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.4	0.6	0.4	0.4	0.3	0.6	0.4	0.3
RESP	0.4	1.0	0.3	0.3	0.4	0.3	0.3	0.3	0.4
Ryb1	0.6	0.3	1.0	0.7	0.5	0.3	1.0	0.5	0.3
R232	0.4	0.3	0.7	1.0	0.4	0.3	0.7	0.5	0.3
Phal	0.4	0.4	0.5	0.4	1.0	0.3	0.5	0.4	0.3
Fail	0.3	0.3	0.3	0.3	0.3	1.0	0.3	0.3	0.3
Fr21	0.6	0.3	1.0	0.7	0.5	0.3	1.0	0.5	0.3
Pepi	0.4	0.3	0.5	0.5	0.4	0.3	0.5	1.0	0.3
EX5b	0.3	0.4	0.3	0.3	0.3	0.3	0.3	0.3	1.0

2.4 Queens

2.4.1 Queen mobility

Queen mobility is a bonus for the number of pseudolegal moves (or alternatively squares attacked) by a queen. Many engines don't have this feature. Others penalise it in the opening, or only count squares not attacked by an enemy pawn/minor/rook (see Hartmann's description of ANALYSE, for instance). Forward mobility, mobility toward the opposing king, etc., are other possible variations.

Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a all give a bonus linear in the number of pseudolegal moves. However, Fruit 2.1 starts the mobility count at -13 , while the Rybkas start it at zero. None of the others include it.

Table 15: Queen mobility

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	1.0	1.0	0.0	0.0	0.8	0.0	0.0
R232	0.0	0.0	1.0	1.0	0.0	0.0	0.8	0.0	0.0
Phal	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.8	0.8	0.0	0.0	1.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.4.2 Queens on 7th rank

This feature concerns bonuses for a queen on the 7th rank. This can be done in PST (though this is not always easy to tell), or directly under certain conditions. Bonuses for secondary majors on the 7th can also apply.

Fruit 2.1 and Rybka 1.0 Beta both give a bonus to a queen on the 7th rank when the enemy king is on the 8th rank, or there is an enemy pawn on the 7th. Rybka 2.3.2a gives a bonus when there is a king or pawn on the 7th or 8th. Crafty 19.0 has the same condition as Fruit/Rybka1, but also requires that the queen be supported by a rook on the 7th. As above, Phalanx XXII counts queens on the 6th/7th/8th, folded into a count of rooks on 7th, with a bonus then applied based on the end result. Pepito 1.59 has no direct bonus to a queen on the 7th, but does increase a rook-on-7th/8th bonus from this. RESP 0.19 similarly only awards a queen on the 7th if a rook is also there. EXChess has a marginal bonus for queens on the 7th in PST, which I chose to ignore. Faile 1.4 also lacks this feature.

Table 16: Queens on 7th rank

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.5	0.7	0.5	0.4	0.0	0.7	0.4	0.0
RESP	0.5	1.0	0.3	0.3	0.4	0.0	0.3	0.6	0.0
Ryb1	0.7	0.3	1.0	0.7	0.3	0.0	1.0	0.3	0.0
R232	0.5	0.3	0.7	1.0	0.3	0.0	0.7	0.3	0.0
Phal	0.4	0.4	0.3	0.3	1.0	0.0	0.3	0.3	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.7	0.3	1.0	0.7	0.3	0.0	1.0	0.3	0.0
Pepi	0.4	0.6	0.3	0.3	0.3	0.0	0.3	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.4.3 Early queen development

This feature penalises a queen for developing too early. Conditions can include how close the queen is to the enemy king, and what “early” means (whether relative to other pieces, or to a move counter).

Crafty 19.0 applies a penalty a queen not on its original square before a side has castled (the comment talks about minor pieces, but seems mistaken). Phalanx XXII penalises a White queen on the 3rd rank or beyond before development is nearly complete; the penalty is linear in how close development is to completing. Pepito 1.59 has a DAMA_DESCOLOCADA penalty that applies when the queen is not on d1 but a minor is on b1/c1/f1/g1, and distance from the queen to the opposing king is greater than 4 squares. Faile 1.4 gives a penalty to a queen that moves before all the minors have moved. Finally, EXchess 5.01beta, RESP 0.19, Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a lack this feature.

Table 17: Early queen development

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.0	0.0	0.4	0.4	0.0	0.3	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.4	0.0	0.0	0.0	1.0	0.4	0.0	0.3	0.0
Fail	0.4	0.0	0.0	0.0	0.4	1.0	0.0	0.6	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.3	0.0	0.0	0.0	0.3	0.6	0.0	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.5 Pawns

2.5.1 Doubled pawns

This feature concerns penalties for doubled pawns. There are multiple ways of doing this, either by punishing each pawn separately (so that tripled pawns are 3x), or the front/rear-most pawn (so that doubled and tripled is the same), or by counting the number of pawns and applying a tabular value (like Crafty). Further variations include a file/rank-based penalty, or an additional penalty for doubled isolated pawns, or doubled pawns on a half-open file.

Fruit 2.1 counts every pawn except the rear-most, while Rybka 1.0 Beta and Rybka 2.3.2 count every pawn except the front-most, possibly with a White/Black switch. In any event, both then give a constant penalty. Crafty 19.0 counts the number of pawns on the file and applies an array-based score. Pepito 1.59 has a `BONUS_DOBLADOS` array which is not used, and applies a constant penalty to each pawn on the file of a doubled pawn. Phalanx XXII applies a constant penalty to a file which has 2 or more pawns (and has an extra penalty for doubled/isolated). Faile 1.4 gives a penalty linear in the number of pawns minus one (so the same as Fruit/Rybka), with the penalty bigger in the endgame. RESP 0.19 gives a constant penalty for every pawn after the first on a file (so the same as Fruit/Rybka/Faile), and then an additional `punishedDb1Iso` penalty when applicable. EXchess 5.01beta penalises every pawn except the first on a file, and furthermore puts such info in a `defects` count, depending on whether it is on the queen/kingside (the latter is probably better included in pawn structure for king safety).

Table 18: Doubled pawns

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.4	0.4	0.4	0.3	0.4	0.4	0.3	0.4
RESP	0.4	1.0	0.7	0.7	0.6	0.7	0.7	0.4	0.7
Ryb1	0.4	0.7	1.0	1.0	0.5	1.0	1.0	0.7	1.0
R232	0.4	0.7	1.0	1.0	0.5	1.0	1.0	0.7	1.0
Phal	0.3	0.6	0.5	0.5	1.0	0.5	0.5	0.5	0.5
Fail	0.4	0.7	1.0	1.0	0.5	1.0	1.0	0.7	1.0
Fr21	0.4	0.7	1.0	1.0	0.5	1.0	1.0	0.7	1.0
Pepi	0.3	0.4	0.7	0.7	0.5	0.7	0.7	1.0	0.7
EX5b	0.4	0.7	1.0	1.0	0.5	1.0	1.0	0.7	1.0

2.5.2 Isolated pawns

This feature concerns penalties for isolated pawns. Variations include counting the number of such pawns and then penalising relative to the white/black counts (Crafty), giving different penalties for half-open/closed files, or possibly a rank/file-based penalty. Counting the number of pawn islands and assessing a penalty based on this would be a related concept.

Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a all give a penalty for an isolated pawn that depends on whether the file is half-open or closed. Crafty 19.0 counts the number of isolated pawns, and the subcount of those on open files, and then applies array-based scores to these. Phalanx XXII gives a file-based penalty, and then adjusts the score based upon the number of knights the opponent has, the number of bishops we have, and whether an opposing rook attacks it. There is then a correction if an isolated pawn is a “ram”, that is, blocked by an enemy pawn face-to-face, and also a doubled-and-isolated penalty. Pepito 1.59 has a file-based array for penalties, though the contents are constant except for the rook files. There is also a further penalty for multiple isolani. Faile 1.4 penalises isolated pawns by a constant amount, with half-open files penalised more (same as Fruit/Rybka). RESP 0.19 also penalises isolated pawns by a constant amount, and gives an additional penalty to isolated pawns that are doubled. EXchess also gives a constant penalty for isolated pawns, and further stores it in a king/queenside defect count.

Table 19: Isolated pawns

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.2	0.4	0.4	0.2	0.4	0.4	0.4	0.2
RESP	0.2	1.0	0.3	0.3	0.3	0.3	0.3	0.3	0.7
Ryb1	0.4	0.3	1.0	1.0	0.2	1.0	1.0	0.3	0.5
R232	0.4	0.3	1.0	1.0	0.2	1.0	1.0	0.3	0.5
Phal	0.2	0.3	0.2	0.2	1.0	0.2	0.2	0.2	0.2
Fail	0.4	0.3	1.0	1.0	0.2	1.0	1.0	0.3	0.5
Fr21	0.4	0.3	1.0	1.0	0.2	1.0	1.0	0.3	0.5
Pepi	0.4	0.3	0.3	0.3	0.2	0.3	0.3	1.0	0.5
EX5b	0.2	0.7	0.5	0.5	0.2	0.5	0.5	0.5	1.0

2.5.3 Backward and/or weak pawns

This feature concerns penalties for backward pawns. There are various definitions of this, and programs differ in what they use (others have a more general “weak” pawn on top of this).

Fruit 2.1 checks first that no adjacent-file pawn is behind or equal to the rank of this one, and then checks to see if by advancing in one move this pawn can meet up with a friendly one, ignoring this latter condition when an opponent’s pawn attacks the advance square. A penalty is then given based upon whether the file is half-open or not. Rybka 1.0 Beta and Rybka 2.3.2a do the same first check, and then see if by advancing one or two squares (not necessarily in one move) whether a pawn can meet a friendly one, again without an enemy pawn intervening by an attack (a minor difference is that Fruit 2.1 checks that a friendly pawn, necessarily doubled, does not block the “backward” pawn’s advance). A penalty is then given based upon whether the file is half-open or not.

Crafty 19.0 calls these “weak” pawns it seems, and the penalty only applies to half-open files (the comment says the penalty is “greater” on such files, but there is no penalty when the file is closed in this version); if a pawn is undefended and when it moves it will be attacked by at least as many enemy pawns as friendly pawns defend it, then it is weak. The penalty can depend on the relative number of attackers/defenders. Isolated pawns are not considered “weak” (the code is skipped in that case, though the comment is a bit confusing). Phalanx XXII requires a backward pawn to be on the 2nd/3rd rank and an enemy pawn a knight’s move in front on an adjacent file, and no friendly pawn adjacent to or guarding the pawn. The penalty then is file-based (as with isolated pawns) with phase adjustments and a dependence on whether an enemy rook attacks the pawn. Isolated pawns can also be backward.

Pepito 1.59 has `PEON_DEBIL` and `PEON_REZAGADO`. If a pawn is `DEBIL`, it is then checked for `REZAGADO` (backward) and `AISLADO` (isolated). The definition of `DEBIL` (weak) is first that it is not defended by a pawn (there is a bonus for being so defended, and also for a “duo” for that matter): then, if there is not an enemy pawn right in front, it is not weak if the square in front has a pawn-defender and is defended as much as attacked (or similarly with a 2-rank push); and then some similar condition when an enemy pawn is right in front. Then “backward” further means a pawn with no friendly pawns behind it and which has more opposing pawn attackers than friendly defenders when it moves forward one square. Isolated can also be backward. There is also an `expuestos` condition (for exposed pawns), which seems to involve whether an enemy pawn is on the file, but if so then `PeonesN` (black pawns) is mistakenly used for both white and black, so I am not sure. The `BONUS_EXPUESTOS_PEP/OTRO` arrays are then applied to the number of `expuestos`, with constant penalties in other cases. There is also `#define EVAL_ATQ_DEBIL` that gives bonuses to pieces that attack “weak” pawns. Not easy to decipher.

In Faile 1.4, backward simply means a pawn behind all other friendly pawns on adjacent files, with isolated a subset of backward. There is a constant penalty assessed based on whether the file is half-open. There is no penalty in the opening phase for a backward pawn on the second rank, unless the pawn is exposed (on a half-open file). In RESP 0.19 backward seems to mean a non-passed pawn that is not isolated and is behind all adjacent pawns. The penalty is constant.

EXchess 5.01beta has backward and weak pawns, with a weak pawn being one that has no adjacent pawn on the same rank or behind it, and a backward pawn being one that additionally is either directly blocked by an opposing pawn, or would come under attack by opposing pawn when moved forward a square. The penalties are constant. I think isolated pawns are also considered weak. If a file does have not a weak pawn, it cannot be considered half-open for the opponent. As with other pawn defects, a king/queenside penalty for shelter is also assessed.

Table 20: Backward and weak pawns

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.4	0.6	0.6	0.5	0.5	0.6	0.4	0.5
RESP	0.4	1.0	0.5	0.5	0.3	0.5	0.5	0.3	0.3
Ryb1	0.6	0.5	1.0	1.0	0.3	0.7	0.7	0.3	0.4
R232	0.6	0.5	1.0	1.0	0.3	0.7	0.7	0.3	0.4
Phal	0.5	0.3	0.3	0.3	1.0	0.3	0.4	0.4	0.4
Fail	0.5	0.5	0.7	0.7	0.3	1.0	0.7	0.3	0.4
Fr21	0.6	0.5	0.7	0.7	0.4	0.7	1.0	0.3	0.4
Pepi	0.4	0.3	0.3	0.3	0.4	0.3	0.3	1.0	0.6
EX5b	0.5	0.3	0.4	0.4	0.4	0.4	0.4	0.6	1.0

2.5.4 Pawn duos

A pawn duo usually refers to two pawns side-by-side. A variation might concern how to score three pawns abreast, with the usual possibilities for rank/file-based changes.

Crafty 19.0 gives a bonus for each pawn in a duo (so three counts 3x). Phalanx XXII gives a bonus for all but the leftmost pawn in a duo, the bonus being larger for pawns on the 4th rank or beyond. It also gives a bonus to a guarded pawn (see next feature), though only one of these can apply. It seems to me that Pepito 1.59 calls something a “duo” if there is a pawn next to it (the method to do this is a bit creative, and uses an horizontal attacks bitboard mask with all bits set), so the same as Crafty 19.0 (including the constancy of the bonus). Next, RESP 0.19 gives a pawn duo bonus to all but the rightmost pawn of a duo, as does EXchess 5.01beta (so three counts 2x). None of Fruit 2.1, Rybka 1.0 Beta, Rybka 2.3.2a, or Faile 1.4 has pawn duos.

Table 21: Pawn duos

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.5	0.0	0.0	0.3	0.0	0.0	0.9	0.5
RESP	0.5	1.0	0.0	0.0	0.7	0.0	0.0	0.5	1.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.3	0.7	0.0	0.0	1.0	0.0	0.0	0.3	0.7
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.9	0.5	0.0	0.0	0.3	0.0	0.0	1.0	0.5
EX5b	0.5	1.0	0.0	0.0	0.7	0.0	0.0	0.5	1.0

2.5.5 Pawn guards

A pawn guard is when one pawn defends another. A variation might concern how to score a doubly guarded pawn. Related to both this and pawn duos might be pawn islands, but I don't think any of the programs examined so far count these.

It seems that only Pepito 1.59 and Phalanx XXII have this term, and the latter does not allow it double-count a pawn in both a guard and a duo.

Table 22: Pawn guards

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.8	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	0.8	0.0	0.0	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.5.6 Pawn centre

This feature gives a bonus to a pawn centre. This is beyond what might be given in PST, and usually depends on their being more than one pawn from a side occupying the centre.

Phalanx XXII gives a bonus to White for pawns on both **d4/e4** that depends upon whether Black opposes this with a pawn on **d5/e5**, with the bonus increased if White has a pawn on one of **c4/f4**. Pepito 1.59 has exactly the same criteria (with slightly different sizing with bonuses).⁶

Table 23: Pawn centre

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

⁶It is unclear to me whether this is more likely to be due to copying or a common source.

2.5.7 Pawn immobility (general)

The pawn immobility penalty notes which pawns cannot move. Some programs take this to mean pawns that are blocked by a unit (if the unit is pawn, possibly split into enemy/own, then the penalty might be different), while others additionally note in the alternative case whether the pawn can advance without being taken by an enemy pawn. Note that immobile/blocked passed pawns are considered elsewhere, and the blocking of pawns at **d2/e2** is also handled separately.

Rybka 2.3.2a gives a constant penalty for a pawn that has some unit on the square in front of it. Phalanx XXII penalises such a pawn with a rank-based bonus, with an additional penalty when this holds up a pawn storm. Isolated pawns that are blocked by pawns have part of their isolation penalty returned. Phalanx XXII also considers pawns that will be captured when they advance, and further folds this into a `cannotmove` variable.

Crafty 19.0 first ignores isolated pawns, and calls a pawn “blocked” if for the next three ranks it will not run into a friendly or opposing pawn and will be sufficiently defended (there could be a logic error in the latter). However, these “blocked” pawns are not really penalised directly, but they are used in some code that tries to keep “lever” possibilities available, with related code for “pawn rams” that is used against humans.

The others lack this feature.

Table 24: Pawn immobility

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.0	0.3	0.3	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.3	0.0	0.0	1.0	0.3	0.0	0.0	0.0	0.0
Phal	0.3	0.0	0.0	0.3	1.0	0.0	0.0	0.0	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.5.8 Blocked central pawns and/or bishops

Blocked central pawns is a feature for when a pawn on **d2/e2** is blocked by some unit on **d3/e3**. Variations include what type of unit the blocking piece is, and perhaps whether the blocked pawn impedes the development of the respective bishop.

Fruit 2.1 and Rybka 1.0 Beta require **wBc1/wPd2** and some unit on **d3**, and penalise the bishop. Crafty 19.0 penalises a pawn on **d2/e2** which is blocked by a minor (of either colour – the bishop only counts when it cannot be driven off by an opponent pawn [this seems buggy]); there is also a more general blocked center pawn criterion in `EvaluateDevelopment` that penalises at most one pawn on **d2/e2** when blocked. Pepito 1.59 has a `PEON_CENTRAL_BLOQUEADO` penalty in the opening that penalises a pawn on **d2/e2** when blocked; there is also a criterion (with the same name) for blocking the **c**-pawn with **Nc3** in a queen-pawn opening, which will be a separate feature.

Phalanx XXII has a “development” penalty for pieces on home squares which is increased for a **Bc1** when **b2/d2** are occupied, but that is not exactly related to the feature here. Phalanx XXII also has a “cannot move” penalty for a pawn on **d2/e2** with some unit right in front of it, meeting the criterion more closely. Faile 1.4 penalises a blocked pawn on **d2/e2** in the opening/middlegame (differing amounts). EXChess 5.01beta has a penalty listed for `BISH_PAWN_BLOCK`, but the penalty is never applied (it seems more likely that this label refers to **c**-pawn blockage). RESP 0.19 also lacks this feature.

Table 25: Blocked central pawns and/or bishops

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.4	0.0	0.7	0.7	0.4	0.7	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.4	0.0	1.0	0.0	0.5	0.5	1.0	0.5	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.7	0.0	0.5	0.0	1.0	0.9	0.5	0.9	0.0
Fail	0.7	0.0	0.5	0.0	0.9	1.0	0.5	1.0	0.0
Fr21	0.4	0.0	1.0	0.0	0.5	0.5	1.0	0.5	0.0
Pepi	0.7	0.0	0.5	0.0	0.9	1.0	0.5	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.5.9 Pawn outpost

Phalanx XXII has something called a “pawn outpost” which doesn’t seem to fit well into any of the other labels. This is a pawn on an “outpost” square (with d4 getting an extra boost), with a bonus being added for each pawn next to the pawn (in a duo), then a bonus added if pawn is guarded by another pawn, with a subtraction if it is attacked by an enemy pawn, and a final bonus if the pawn is guarded by pieces. The bonus is only applied in the middlegame.

Perhaps this could be considered with guards/duos, but it seems to be distinct, and in particular depends on the pawn being on an “outpost” square. None of the other engines have such a feature.

Table 26: Pawn outpost

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.5.10 Pawn/piece interaction?

I had some thought to try to measure the general “pawn/piece interaction” of engines (for some it is quite low, while others like Phalanx XXII do a lot in this regard), but decided it was too difficult to include well. Pepito 1.59 also has many bonuses for pieces attacking the opponent’s weak pawns.

2.6 Passed and candidate pawns

2.6.1 Candidate passed pawns

The candidate passed pawns feature concerns said pawns. Variations can include the definition and the scaling of the bonus. It is assumed in the below that the pawn is not truly passed.

Fruit 2.1 requires a pawn to be unopposed, and then counts the number of possible defenders and attackers (friendly pawns behind or abreast, and enemy pawns in front). If there are at least as many defenders as attackers, Fruit 2.1 then further checks that the difference between “direct” defenders and attackers is again nonnegative. A bonus is then awarded via a rank-based 10-30-60-100 scaling (including the irrelevant 7th rank, for a pawn there must be passed). Rybka 1.0 Beta and Rybka 2.3.2a also require a pawn to be unopposed and count the number of possible defenders and attackers in the same manner, but do not fret about the direct attackers/defenders. The bonus is also a rank-based 10-30-60-100 scaling (again including the irrelevant 7th rank).

Crafty 19.0 first checks that there are no opposing pawns on the file (including behind the pawn), that the pawn is not isolated, and that the opponent is not directly attacking it. A loop over squares in front of the pawn is then made; if a (friendly) pawn is hit, or a square the enemy attacks is reached, the number of attackers/defenders is computed, and if the number of defenders is at least as large and the pawn will be passed when it passes this square, then the pawn is a candidate. There is then (possibly buggy?) code that checks if the zeroth file (I think this is the **a**-file) has a candidate, and if not, then splits into two cases depending upon whether the pawn is on the king/queenside; on the queenside it checks for friendly pawns on both of the next two rightward files, and if the opponent has no pawns on the file in question, no pawns on either the file two or three to the right, and no more than two pawns on the file to the right, then a pawn is a candidate. I think an example is White having pawns on the **def**-files, Black has no pawns on the **dfg**-files, and no more than two pawns on the **e**-file. Finally, there is code for “hidden” passed pawns, a simple case of which is **wPb5/Pa6** and **bPa7**. A direct rank-based score is given for this (only nonzero on 5th/6th ranks), while the files of the candidates are simply catalogued, and bonuses for outside pawns and spreads are added later.

Pepito 1.59 defines a candidate by a pawn that is at least on the 4th rank, there is no enemy pawn in front on this file, there is at most one adjacent file which has an enemy pawn in front (so edge files are automatically OK), and that the enemy has at most one pawn in front on said file. A constant bonus is then given, with it increased fourfold when the side with fewer candidate pawns has no pieces. There is also a **PEON_CUASI_PASADO** variable, but it is unused.

It seems that Phalanx XXII, EXchess 5.01beta, RESP 0.19, and Faile 1.4 lack this feature.

Table 27: Candidate passed pawns

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.3	0.3	0.0	0.0	0.4	0.2	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.3	0.0	1.0	1.0	0.0	0.0	0.8	0.4	0.0
R232	0.3	0.0	1.0	1.0	0.0	0.0	0.8	0.4	0.0
Phal	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.4	0.0	0.8	0.8	0.0	0.0	1.0	0.4	0.0
Pepi	0.2	0.0	0.4	0.4	0.0	0.0	0.4	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.6.2 Passed pawns, initial bonuses

It is not exactly easy to determine how to split up bonuses given for passed pawns. The current feature applies to those given based upon “static” considerations, such as whether a passed pawn is protected, connected, or outside (distance to kings is considered below).

Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a give a rank-based bonus based on a 10-30-60-100 scaling. Crafty 19.0 gives no explicit “standard” bonus, but folds these into bonus for outside pawns passed on the number of opposing pieces. A bonus for connected passed pawns is also added, with this being increased when both reach the sixth rank. Phalanx XXII starts with a rank-based bonus (10-15-25-40-60-95), and triples this when the pawn is either guarded by another pawn or in a duo (penalties for the pawn being blocked can be applied before this tripling, but those are considered below). An outside bonus is also given, and depends on the material situation. Pepito 1.59 gives a small initial bonus (5cp), and then a rank-based bonus (5-15-30-55-75-100) which is multiplied by a factor based on the material, and may be decreased if the pawn is blocked (see below). There is then a rank-based `PASADO_FUERTE` bonus which applies unless a passed pawn is “weak” by prior definition (in which case there can still be a constant `DONDE_CREES_Q_VAS` bonus). Connected passed pawns on the sixth or seventh rank get a huge bonus when the opponent has a piece left. There is finally an `alejados` (away) bonus for outside pawns, which is dependent on material.

Failé 1.4 gives a square-based bonus (triple the PST in the endgame) for a passed pawn, and an extra bonus if it is not isolated. RESP 0.19 gives a rank-based bonus (8-14-30-60-100-130) multiplied by a game-stage factor, and a constant bonus for sufficiently far advanced connected passed pawns whose ranks differ by no more than 1 (again multiplied by a game-stage factor), and finally an adjustment for outside passed pawns. EXchess 5.01beta gives a standard bonus linear in the rank, and linear in the stage unless it is protected by a pawn, in which case the maximal stage is always used. Connected passed pawns beyond the 4th rank receive a stage-based bonus, and outside pawns (measured by more than three files from the opposing king) get a bonus again linear in both the stage and rank.

Table 28: Passed pawns, initial bonuses

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.4	0.3	0.3	0.4	0.3	0.3	0.2	0.4
RESP	0.4	1.0	0.4	0.4	0.4	0.3	0.4	0.5	0.5
Ryb1	0.3	0.4	1.0	1.0	0.4	0.3	1.0	0.3	0.3
R232	0.3	0.4	1.0	1.0	0.4	0.3	1.0	0.3	0.3
Phal	0.4	0.4	0.4	0.4	1.0	0.2	0.4	0.5	0.3
Fail	0.3	0.3	0.3	0.3	0.2	1.0	0.3	0.3	0.3
Fr21	0.3	0.4	1.0	1.0	0.4	0.3	1.0	0.3	0.3
Pepi	0.2	0.5	0.3	0.3	0.5	0.3	0.3	1.0	0.3
EX5b	0.4	0.5	0.3	0.3	0.3	0.3	0.3	0.3	1.0

2.6.3 Passed pawns, support/attack by kings

This feature refers to whether the kings are far/close to the passed pawn. There is some overlap with the “outside” concept, though not every engine puts king location into the computation of that.

Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a give a rank-based bonus/penalty (with 10-30-60-100 scaling) times the distance of the square in front of the pawn to the respective kings. Crafty 19.0 gives a rank-based bonus (12-60-100) when the friendly king is on the same or an adjacent file and is not behind the pawn. Phalanx XXII gives an endgame bonus essentially linear in the distance of the enemy king to the square in front of the pawn plus the rank, and an additional bonus if the friendly king attacks the square in front of the pawn. Pepito 1.59 gives a rank-based bonus (5-20-60-100, based on the rank of the king) if the king guards the pawn. When the opponent has one piece, the `PASA_ALEJA_REY` array gives a bonus for the opposing king not being close to the passed pawn.

EXchess 5.01beta, RESP 0.19 and Faile 1.4 lack this feature.

Table 29: Passed pawns, king locations

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.3	0.3	0.4	0.0	0.3	0.4	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.3	0.0	1.0	1.0	0.5	0.0	1.0	0.4	0.0
R232	0.3	0.0	1.0	1.0	0.5	0.0	1.0	0.4	0.0
Phal	0.4	0.0	0.5	0.5	1.0	0.0	0.5	0.3	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.3	0.0	1.0	1.0	0.5	0.0	1.0	0.4	0.0
Pepi	0.4	0.0	0.4	0.4	0.3	0.0	0.4	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.6.4 Passed pawns, freedom to advance

This feature regards bonuses/penalties for when passed pawns are free to advance or are blocked. I decided to put supporting rooks in this category.

Fruit 2.1 gives a constant bonus to a passed pawn for which the square in front of it is empty, and for which SEE says the pawn can advance safely. Rybka 1.0 Beta and Rybka 2.3.2a have a rank-based bonus (10-30-60-100 scaling) first for the pawn not having any friendly unit anywhere in front of it, then another similar bonus for the pawn not having any enemy unit anywhere in front of it, and a third such bonus if every square in the path of the pawn is defended at least as much as it is attacked. Crafty 19.0 gives a constant bonus to a friendly rook behind a passed pawn (halved if the pawn is blocked by an enemy piece), and a bonus to a rook behind an enemy passed pawn. A rank-based bonus (8-12-16-30-45-60) is given to a piece that blockades an enemy passed pawn. Phalanx XXII reduces the standard bonus for a passed pawn by half when the square in front of it is occupied by an enemy piece, and by a quarter when the square in front of that is so occupied (both can apply). If the square in front of the pawn is empty, SEE is used for a pawn on the 7th rank, and a bonus given if the pawn can advance safely; if the pawn is on the 6th and the queening square is empty, SEE is used on both the forward squares. All bonuses are constant. A rook (either friendly or enemy) behind a passed pawn is rewarded. There is also a condition with the `hung` variable (for hung pieces) for a pawn on the 6th/7th rank that is not blocked.

Pepito 1.59 has a constant bonus for a friendly rook that is on the same file as a passed pawn (in either direction), and an unused `TORRE_CONTRA_PASADO` variable (presumably for enemy rooks). It also reduces the standard rank-based bonus when the square in front of the pawn is occupied. RESP 0.19 gives a rank-based bonus (2-4-6-12-25-32, also dependent on stage) to a knight that blocks a passed pawn. Faile 1.4 and EXchess 5.01beta lack this feature.

Table 30: Passed pawns, freedom to advance

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.2	0.4	0.4	0.3	0.0	0.3	0.4	0.0
RESP	0.2	1.0	0.3	0.3	0.2	0.0	0.2	0.2	0.0
Ryb1	0.4	0.3	1.0	1.0	0.4	0.0	0.3	0.2	0.0
R232	0.4	0.3	1.0	1.0	0.4	0.0	0.3	0.2	0.0
Phal	0.3	0.2	0.4	0.4	1.0	0.0	0.5	0.3	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.3	0.2	0.3	0.3	0.5	0.0	1.0	0.3	0.0
Pepi	0.4	0.2	0.2	0.2	0.3	0.0	0.3	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.6.5 Unstoppable pawns, races, pawn ending evaluators

This feature involves unstoppable pawns (usually when the opponent has only a king), pawn races, and I also chose to include pawn ending evaluators here, as usually those are heavily based on passed pawns.

Fruit 2.1 and Rybka 1.0 Beta have only a constant “unstoppable” bonus when the opponent has only a king left. Here unstoppable depends on the “square” of the pawn or whether the friendly king shepherds it. Fruit 2.1 additionally has a KPK/KKP recogniser. Rybka 2.3.2a has a special module for pawn endgame evaluation, the major feature of which seems to be a bonus of 11200/3717 pawns times the difference between ranks of foremost unstoppable passed pawns (if any), with unstoppable as before. Most of the “normal” passed pawn bonuses (like king distance) are not used in this routine. Crafty 19.0 has an `EvaluatePassedPawnRaces` routine that is used whenever one side has no piece and the opponent has a passed pawn. The routine first checks for KPK (though the logic seems to allow any number of pawns for the superior side). It then computes the fastest “queener” for each side, and returns a value based on this; if one side is two moves or more faster, the bonus is linearly rank-based after a constant bonus, and if the race is closer, an additional condition regarding queening with check is made. The normal routines for passed pawns are then additionally used.

Phalanx XXII also has special code for pawn endgames, containing specific code for the `onepawn` case and also code for races. In general, pawn mobility and king activity are stressed, and the bonuses regarding elements like enemy king distance and outside passers are copied over. Pepito 1.59 gives a bonus for having the opposition in a pawn endgame, and has some racing code when a side has one or (particularly) no pieces. RESP 0.19 has uncatchable pawn eval code, which computes the number of steps to queening for each side and gives a bonus if the race is sufficiently one-sided. There is also a KPK recogniser. EXchess 5.01beta computes a taxi-cab distance and gives passed pawns a bonus linear in the rank if they cannot be caught; it also has an `OPPOSITION` variable that is unused. Faile 1.4 seems to lack this feature.

Table 31: Passed pawns, sundry

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.6	0.3	0.5	0.7	0.0	0.5	0.4	0.3
RESP	0.6	1.0	0.3	0.4	0.5	0.0	0.4	0.3	0.3
Ryb1	0.3	0.3	1.0	0.4	0.4	0.0	0.7	0.3	0.3
R232	0.5	0.4	0.4	1.0	0.3	0.0	0.4	0.3	0.4
Phal	0.7	0.5	0.4	0.3	1.0	0.0	0.5	0.4	0.3
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.5	0.4	0.7	0.4	0.5	0.0	1.0	0.3	0.3
Pepi	0.4	0.3	0.3	0.3	0.4	0.0	0.3	1.0	0.3
EX5b	0.3	0.3	0.3	0.4	0.3	0.0	0.3	0.3	1.0

2.7 King danger

2.7.1 King danger, when to use

This feature concerns material and/or phase conditions which control the consideration of king danger (either by pieces or for pawn shelter/storm), and also castling aspects to the extent that they pertain to this (castling in general is considered in development).

Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a ignore all king danger and shelter/storm unless the opposing side has a queen and another piece. Fruit 2.1 and Rybka 1.0 Beta can reduce the shelter/storm score (via an averaging) when the possibility of castling exists. In each of these three, there is a general decrease of scores in the endgame via the interpolation (which is a separate feature). Crafty 19.0 has an `EG_MAT` variable set to 14, and if either side has no more material than this, then some aspects of king safety are skipped. When the opposing side lacks a queen, any shelter/storm aspects are halved. These aspects are “tempered” and included both by themselves (when there is enough material) and folded into the final tropism-based penalty. Finally, if castling is possible, prospective defects are included into the computation.

It seems to me that Phalanx XXII has no explicit condition for this, but rather only counts king safety elements in its “middlegame” score (the split of middle/endgame is approximately that the first third is the middlegame, the second uses interpolation, and the last third is the endgame). Wing pawn pushes are penalised when castling rights exist. Pepito 1.59 has various king security penalties, and it seems to me that the tropism-based ones are always used (there is, however, an augmentation array multiplier based on material). Pawn defects are halved when the opponent lacks a queen (and these are folded into the final computation from the tropisms). The `calcseg` flag requires that the opponent’s material be more than 12 and have more than 4 pawn defects (in which case contact squares can also be considered). Castling rights are folded into pawn defects.

Faille 1.4 has a general switch in PST for kings in the endgames (other engines also have this), whose usage depends on there being less than 5 pieces on the board, but I ignore this here. There are no “dynamic” king safety considerations in any case. RESP 0.19 ignores king safety when one of the kings is on the opponent’s 7th/8th rank. The `king_safety_score()` function is called in the opening and early/late middlegame stages, but not the endgame (either 6 or less pawns or pieces in total). There is no adjustment for future castling homes. EXchess 5.01beta ignores pawn defects in the endgame (either 6 or less pawns or pieces in total). Defects from both board sides are counted if one has not castled. In order for contact squares to be computed for the enemy king, one must have a queen near it, or a half-open file next to it with a major on it.

Table 32: King danger, when to use

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.3	0.4	0.3	0.4	0.0	0.4	0.5	0.3
RESP	0.3	1.0	0.3	0.3	0.3	0.0	0.3	0.3	0.4
Ryb1	0.4	0.3	1.0	0.8	0.5	0.0	1.0	0.5	0.4
R232	0.3	0.3	0.8	1.0	0.4	0.0	0.8	0.3	0.3
Phal	0.4	0.3	0.5	0.4	1.0	0.0	0.5	0.4	0.4
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.4	0.3	1.0	0.8	0.5	0.0	1.0	0.5	0.4
Pepi	0.5	0.3	0.5	0.3	0.4	0.0	0.5	1.0	0.4
EX5b	0.3	0.4	0.4	0.3	0.4	0.0	0.4	0.4	1.0

2.7.2 King danger from pieces, method

This feature relates to the method used to compute king danger from pieces. Popular methods include distances (with tropisms), contact squares (either square-based or piece-based), and checking possibilities.

Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a use contact squares, counting each piece that attacks a square around the enemy king. Fruit 2.1 allows more than one pawn to be counted. Crafty 19.0 uses tropisms (weighted distances from a piece to the enemy king, with adjustments for things like rooks and open files) and folds these into a array. There is also a bonus for `QUEEN_IS_STRONG` when the opponent has many more pawn defects, and one with tropisms for an offside queen. Phalanx XXII has a method that involves looking at all squares around the king, determining whether the opponent attacks it (when a bonus is added), and if so and we don't defend the square, then an extra bonus is added, especially if it is the queen that attacks it. Another bonus for safe checks to such squares is added. There are also (direct) tropism bonuses, some of which are blended into pawn storms. Pepito 1.59 has both tropisms and contact squares, with the latter being computed as with Phalanx XXII, except the safe checks are not considered. Tropisms for rooks on half-open files are modified slightly, as are bishop tropisms when there is already a contact square from it. RESP 0.19 has a simple tropism method that gives a bonus (quadratic for the queen and linear for other pieces) based on the distance to the enemy king when it is sufficiently small. EXchess 5.01beta uses contact squares, counting whether each square is attacked. Faile 1.4 seems to lack king danger from pieces.

Table 33: King danger from pieces, method

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.7	0.3	0.3	0.3	0.0	0.3	0.6	0.3
RESP	0.7	1.0	0.3	0.3	0.3	0.0	0.3	0.5	0.3
Ryb1	0.3	0.3	1.0	1.0	0.4	0.0	0.9	0.4	0.5
R232	0.3	0.3	1.0	1.0	0.4	0.0	0.9	0.4	0.5
Phal	0.3	0.3	0.4	0.4	1.0	0.0	0.4	0.7	0.5
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.3	0.3	0.9	0.9	0.4	0.0	1.0	0.4	0.5
Pepi	0.6	0.5	0.4	0.4	0.7	0.0	0.4	1.0	0.6
EX5b	0.3	0.3	0.5	0.5	0.5	0.0	0.5	0.6	1.0

2.7.3 King danger from pieces, relative weightings

This feature concerns the relative weight given to the type and number of attackers. For instance, some programs count queen attacks as more important than attacks from minor pieces. Similarly, some programs have a large “multiplier” effect when there are multiple attackers of the enemy king.

Fruit 2.1 counts attackers constantly (including pawns), and then weights them as (0:1:2:4) for pawns/minors/rooks/majors, and then multiplies the weight-sum by a table-lookup based on the number of attackers (0-128-192-224-240... , starting at one attacker). Rybka 1.0 Beta and Rybka 2.3.2a count attackers constantly (with at most one pawn attacker being counted) and then weight them as (0:941:418:666:532) for PNBQR, and then multiply the weight-sum by a table-lookup for attackers (0-30-57-81-81... for Rybka 2.3.2a, similar for Rybka 1.0 Beta).

Crafty 19.0 has slightly larger tropism values for queens and rooks compared to minors, but it is not much. The final tropism table starts at -25 , reaches 0 at 8 tropisms, 40 at 16 tropisms, 120 at 24 tropisms, 145 at 32 tropisms, 180 at 36 tropisms, at which point it becomes constant. It seems that the only piece-type adjustment Phalanx XXII makes is for queens on a contact square that is not defended and attacked more than once. The `khung` value (which also depends on pawn storms/shields) is squared if it exceeds 1. The tropism counts of Pepito 1.59 are rather like Crafty’s. The final table starts at 0, and is approximately linear up to 30 for 12 tropisms. The security count for contact squares is as with Phalanx XXII, with the overall score being linear in this (there is also a material-based diminishment).

RESP 0.19 weights attackers as (6:5:6) for minors/rooks/queens, and as noted above the bonus is linear in distance except for queens for which it is quadratic. There is no specific adjustment for additional attackers given. EXchess 5.01beta gives a constant `KING_ATTACKS` bonus for each contact square that is attacked. Faile 1.4 lacks this feature.

Table 34: King danger from pieces, relative weightings

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.3	0.2	0.2	0.4	0.0	0.4	0.6	0.3
RESP	0.3	1.0	0.3	0.3	0.5	0.0	0.3	0.4	0.3
Ryb1	0.2	0.3	1.0	1.0	0.3	0.0	0.3	0.3	0.3
R232	0.2	0.3	1.0	1.0	0.3	0.0	0.3	0.3	0.3
Phal	0.4	0.5	0.3	0.3	1.0	0.0	0.3	0.4	0.4
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.4	0.3	0.3	0.3	0.3	0.0	1.0	0.4	0.3
Pepi	0.6	0.4	0.3	0.3	0.4	0.0	0.4	1.0	0.4
EX5b	0.3	0.3	0.3	0.3	0.4	0.0	0.3	0.4	1.0

2.7.4 King shelter from friendly pawns

This feature concerns methods (and relative scaling) for king shelter, and also fianchettoed bishops to the extent that they are related to this. Back rank considerations are another element here.

Fruit 2.1 gives a penalty according to the first pawn on a file that the king is adjacent to or on. The penalty is doubled for the file of the king. The rank-based penalty is (0:11:20:27:32:35:36), the last being when there is no pawn on the file. There is a “back-rank” penalty of 11 if the total penalty would otherwise be 0. Rybka 1.0 Beta and Rybka 2.3.2a penalise based on 4x3 grids of pawns depending on the king-file location. There are three possibilities, the choices being the **a-c**, **d-f**, or **f-h** files. As with Fruit 2.1, the penalty is doubled for the file of the king. The rank-based penalty (based on the first pawn) is (0:214:749:945:1121) in Rybka 1.0 Beta, with 794 being given as a back-rank penalty.

Crafty 19.0 first gets a score based solely on the king square, then adds in the defect score based on whether the king is on the king/queenside. This defect score is a folded computation mostly based upon missing pawns on the 2nd/3rd rank plus open and half-open files (for each side) on the relevant board side, and also has a bonus for a stonewall formation. There is a possible **BACK_RANK** penalty. an additional king safety penalty for kings on opposite wings, and additional penalties for a king in the center with open files around. Finally, there is a “trojan horse” check and a **g2/b2** mate detection when the opponent has a pawn on **f3/c3**, the latter of which can be offset if there is a fianchettoed bishop.

Phalanx XXII has shield bonuses for pawns/bishops next to the king on either side, with an increased bonus if there is a pawn in front of such a bishop. For the square in front of the king, a bonus twice as large is given for a pawn, which is increased if there is another pawn on either side of that pawn. If a bishop is in front of the king, the shield is incremented, with another addition if there is a pawn on either side of it, and if there is, yet another shield-point is added if a pawn is front of the bishop. Another increment for a pawn two squares in front of the king, and again doubled if there is a neighboring pawn. Finally, pawns directly in front on adjacent files to the king are rewarded. Most of the first few shield-increments have direct bonuses also. If the opponent has a queen or two rooks, then files lacking friendly pawns that are adjacent to the king are penalised. Another penalty is when the king is in the center and the opponent has a half-open file (increased for an open file). The shield score linearly affects the **khung** variable, and the overall result is quadratic in the latter.

Pepito 1.59 uses one of three choices of files depending on which side the king castled. Defects in the center are solely based on whether the adjacent files have friendly/enemy pawns (that is, open/half-open files). For the kingside, the **f**-file is checked for pawns, with a double penalty if the opponent has a rook on it; a lack of a pawn at **f2** is penalised half as much. A similar computation is done for the **g**-file with no pawn at **g2/g3** being penalised unless a fianchetto bishop is on **g2** (when overall defects are halved), and a mating possibility with **g2** is also considered. The **h**-file concerns openness and then a pawn at **h2/h3**. If there is little opposing material and no pawn defects, a bonus is given. The defects are then penalised via a count-based table.

Faille 1.4 penalises linearly in the distance from the king to the backmost pawn on the three files around the king, with a very marginal increased multiplier for the file of the king. RESP 0.19 gives a penalty depending on whether the pawn wall is critical (0-3 plusses), bad (4-5), medium (6), or good (7). The penalties are (55:35:18:4). Four points of safety are given for a pawn directly in front of the king, and otherwise two for a pawn two squares in front, and similarly on the adjacent files with the points halved for each. EXchess 5.01beta has variables for QUEEN/KING_SIDE_DEFECTS, but they are unused. The code computes pawn defect scores for each board side, with missing pawns on the 2nd rank demerited, and then half as much again on the 3rd rank. The pawneval code also counts defects for doubled/isolated/backward pawns. The score adjustment is linear in these. The penalties are file-based, though the variation is small.

Table 35: King shelter from friendly pawns

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.4	0.3	0.3	0.3	0.3	0.3	0.6	0.4
RESP	0.4	1.0	0.4	0.4	0.4	0.3	0.4	0.3	0.3
Ryb1	0.3	0.4	1.0	1.0	0.3	0.4	0.6	0.3	0.3
R232	0.3	0.4	1.0	1.0	0.3	0.4	0.6	0.3	0.3
Phal	0.3	0.4	0.3	0.3	1.0	0.3	0.3	0.3	0.4
Fail	0.3	0.3	0.4	0.4	0.3	1.0	0.4	0.3	0.3
Fr21	0.3	0.4	0.6	0.6	0.3	0.4	1.0	0.3	0.3
Pepi	0.6	0.3	0.3	0.3	0.3	0.3	0.3	1.0	0.3
EX5b	0.4	0.3	0.3	0.3	0.4	0.3	0.3	0.3	1.0

2.7.5 King storm from enemy pawns

This feature concerns king storm potential from enemy pawns, and also half-open (or open) files around the king (there is some overlap with rooks here, though usually the penalty here does not depend on the existence of a rook or queen). One typical factor is whether kings are castled on pposite wings.

Fruit 2.1 uses a system of adjacent files for the opposing king, giving a storm bonus of (1:3:6) for file-leading pawns on the 4th rank and beyond. Rybka 1.0 Beta and Rybka 2.3.2a do similarly, with (310:653:2334) as the numerology in Rybka 1.0 Beta. Phalanx XXII has a large amount of storm code, some of which only applies when kings are on opposite sides of the board. In that case it penalises blocked pawns (blocked in any way), then gives pawns a square-based bonus. There is some overlap with piece tropism, some of which can change the `khung` count (usually only when kings are on opposite wings), and others of which modify the score directly. Faile 1.4 considers storms when the king files differ by more than 2, and gives a bonus linear in the rank for the three adjacent files to the opposing king, with another bonus for half-open files. Pepito 1.59 has PST-based `KING/QUEEN_STORM` arrays which take the place of `TABLE_MED/FIN` in the case of opposite wing castling. EXchess 5.01beta and RESP 0.19 seem not to have any pawn storm considerations. Crafty 19.0 can change the king PST if there are no pawns on one half of the board, and also has an asymmetric `CASTLE_OPPOSITE` bonus. Neither of these seems exactly like a pawn storm.

Table 36: King storm from enemy pawns

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	1.0	1.0	0.2	0.4	0.8	0.3	0.0
R232	0.0	0.0	1.0	1.0	0.2	0.4	0.8	0.3	0.0
Phal	0.0	0.0	0.2	0.2	1.0	0.3	0.2	0.3	0.0
Fail	0.0	0.0	0.4	0.4	0.3	1.0	0.4	0.3	0.0
Fr21	0.0	0.0	0.8	0.8	0.2	0.4	1.0	0.3	0.0
Pepi	0.0	0.0	0.3	0.3	0.3	0.3	0.3	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.8 Conditions regarding game phase

2.8.1 Game phases and/or interpolation

Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a use a system where the minors, rooks, and queens are related in the ratio 1:2:4. Two scores are computed, and the result interpolated from these depending on the phase. The interpolation is linear in Fruit 2.1 and Rybka 2.3.2a (it is table-based in the latter, but the table is simply linear), and is somewhat nonlinear for some phases in Rybka 1.0 Beta. Crafty 19.0 has some `EvaluateWinner` code that kicks in when there is sufficiently little material left, but this seems not that relevant to the feature here. Phalanx XXII computes two scores, and (linearly) interpolates between them in essentially the middle third of the game, with the phase based on (100:350:550:1050). Faile 1.4 has three separate evaluation functions, with opening for 12+ pieces, middle for 5-11, and endgame for 4 or less. RESP 0.19 has four evaluation functions, with the opening for 13+ pieces and 13+ pawns, the early midgame for at least 12 pieces or pieces and pawns both exceeding 10, and the late midgame for more than 6 each of pawns and pieces. EXchess 5.01beta also has four phases (though just one function), with 14+ pieces and pawns each for the opening, 8+ each for early midgame, and 6+ each for late midgame. Pepito 1.59 has three phases (`APERTURA` is unused) which indicate various items to include, and has various elements that are material-dependent. For instance, there is a king endgame `PST` whose weight is given by the `AUMENTA` array based on the amount of opposing material. The definition of `MEDIO_JUEGO_INI` requires more than 8 total pawns and 7 total pieces, and `MEDIO_JUEGO_FIN` reduces these to 6 and 5 respectively.

Table 37: Game phases and/or interpolation

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	1.0	0.3	0.3	0.3	0.6	0.3	0.4	0.7
Ryb1	0.0	0.3	1.0	0.8	0.6	0.3	0.8	0.3	0.3
R232	0.0	0.3	0.8	1.0	0.7	0.3	1.0	0.3	0.3
Phal	0.0	0.3	0.6	0.7	1.0	0.3	0.7	0.3	0.3
Fail	0.0	0.6	0.3	0.3	0.3	1.0	0.3	0.4	0.6
Fr21	0.0	0.3	0.8	1.0	0.7	0.3	1.0	0.3	0.3
Pepi	0.0	0.4	0.3	0.3	0.3	0.4	0.3	1.0	0.3
EX5b	0.0	0.7	0.3	0.3	0.3	0.6	0.3	0.3	1.0

2.8.2 General development and/or thematic moves

This feature concerns code for general development (other than simply a yes/no for castling, and avoiding early queen development, both of which are considered elsewhere) and thematic moves such as not blocking the c-pawn in queen-pawn openings.

Crafty 19.0 penalises $wPc2/wBnc3/wPd4$ when there is no white pawn on $e4$. Phalanx XXII has a bonus for $wPc4/bPd5$, and increases this when there is additionally $wPd4/bPe6$ or $bPf5$. There is also a penalty for $wPc2$ with $c3$ occupied, and also one for $wPc2$ if there is a white pawn at $e2/e3$ or a black pawn at $e6/c6$. There is also a general development count that consists of a possible of 29 demerits, 2 for each rook, up to 7 for a bishop, 1 for the queen, and 5 each for the knights, which is folded into a formula. Pepito 1.59 penalises $wPd4/bPd5$ when accompanied by $wPc2/wNc3$.

RESP 0.19, EXchess 5.01beta, Faile 1.4, Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a lack this feature.

Table 38: General development and/or thematic moves

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.0	0.0	0.3	0.0	0.0	0.6	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.3	0.0	0.0	0.0	1.0	0.0	0.0	0.3	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.6	0.0	0.0	0.0	0.3	0.0	0.0	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.8.3 Bonuses/penalties with castling

This feature corresponds to whether there are bonuses/penalties for castling as pertains to development. It does not consider shelter/storm or more specific king safety measures therein.

Crafty 19.0 penalises a move that loses castling rights (doubling it if the opponent has a queen), and generally gives a penalty for not having castled. Phalanx XXII has a system where the openness of the e-file is combined with the number of pieces in the way of castling to derive a penalty. Faile 1.4 gives a bonus for having castled and gives a penalty for a king that has otherwise moved, increasing the latter when the file of the king has no sheltering pawn. RESP 0.19 gives a penalty for losing all castling rights. EXchess 5.01beta gives a bonus in the opening and early middlegame for having castled, and a penalty in the opening for losing all castling rights. Pepito 1.59, Fruit 2.1, Rybka 1.0, and Rybka 2.3.2a lack this feature.

Table 39: Bonuses/penalties with castling

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.4	0.0	0.0	0.2	0.5	0.0	0.0	0.6
RESP	0.4	1.0	0.0	0.0	0.2	0.4	0.0	0.0	0.5
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.2	0.2	0.0	0.0	1.0	0.3	0.0	0.0	0.2
Fail	0.5	0.4	0.0	0.0	0.3	1.0	0.0	0.0	0.7
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EX5b	0.6	0.5	0.0	0.0	0.2	0.7	0.0	0.0	1.0

2.8.4 Bishop/knight ending advantage

This feature regards giving a bishop a bonus over a knight in an endgame, or perhaps penalising a knight if there are widespread pawns. It is possible to do the former (either relatively or in an absolute sense) via a material table, though I did not investigate this too thoroughly for the Rybkas. I guess it is also possible to do this via PST if that is phase-based.

Crafty 19.0 gives a `BISHOP_OVER_KNIGHT_ENDGAME` bonus when the side with the bishop has no other pieces except possibly another minor, the opposing side has no bishop, and there are pawns that intersect both the `abc`- and `fgh`-files. Phalanx XXII has an endgame demerit to a knight when there are widespread pawns (file distance), with the penalty increasing when the total nonpawn material is not more than two minors. Pepito 1.59 has a `CABALLO_EN_FINAL` penalty when the number of pieces is equal and less than 4 for each side, and there are pawns on each side of the board. The penalty is linear in the number of pieces left.

EXchess 5.01beta, RESP 0.19, Faile 1.4, and Fruit 2.1 lack this bonus, as do Rybka 1.0 Beta and Rybka 2.3.2a unless such information is encoded in the material imbalance tables.

Table 40: Bishop/knight endgame advantage

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.0	0.0	0.3	0.0	0.0	0.5	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.3	0.0	0.0	0.0	1.0	0.0	0.0	0.5	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.5	0.0	0.0	0.0	0.5	0.0	0.0	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.8.5 Special endgame code

This feature says whether an engine has specific code for various endgames. Pawn endgames are ignored here, as they are usually more dependent on elements of passed pawns. This feature is also different from draw recognition (in N versus P for instance – the case of RP vs R could be in either feature, depending on what is done), where just the result is found. The most typical example for this feature is checkmate with KBN vs K.

Crafty 19.0 has special code to ensure the king is forced to the proper corner in KBN vs K, and otherwise simply forces the weaker king to the side of the board. Phalanx XXII has special code for mating with a knight and bishop, and some RP vs R code. Pepito 1.59 has special code for BN and BB.

EXchess 5.01beta, RESP 0.19, Faile 1.4, Fruit 2.1, Rybka 1.0 Beta, and Rybka 2.3.2a lack this feature.

Table 41: Special endgame code

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.0	0.0	0.5	0.0	0.0	0.5	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.5	0.0	0.0	0.0	1.0	0.0	0.0	0.5	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.5	0.0	0.0	0.0	0.5	0.0	0.0	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.8.6 Draw recognition, insufficient material

This feature is about draw recognition in endgames. In some cases, this could be done via a material table.

Fruit 2.1 handles four-unit setups, some with recognisers and others with reductions (like R vs N), the latter via a more general code when a side is less than a minor up and has two or less majors. There is special code for KBP vs K, and a note that being one minor up with no pawns around is not likely to win. The case of BB vs N is “barely drawish”, while NN vs P is considered almost always drawn. There is specific code for draw recognition in pawn-versus-piece endgames, for KNP vs K draws, for RP vs R and BP vs B, and also code to see if the stronger side has pawns on more than one file (used with blind bishops).

Rybka 1.0 Beta and Rybka 2.3.2a handle draw recognition via adjustments in the material table. There is no special code for blind bishops (they are given rather high scores). It is not easy to briefly summarise the table adjustments, but NN vs P is thought to be a draw, being an exchange up with no queens or pawns is drawish, being up a minor with no pawns is drawish unless a queen and a rook is held, and the basic draws like two knights or a minor versus a bare king are included. Finally, BB vs N by itself is thought winnable.

Crafty 19.0 has an `EvaluateWinner` routine that handles four-unit setups (like KR vs KN is not likely to be a win). There is also code for blind bishops, while BB vs N is noted as winnable with any other two minors versus one minor configuration being drawn. Phalanx XXII has code for a minor piece versus a pawn or minor, but doesn’t seem to know that a minor by itself cannot win(?!). Blind bishops are handled in the minor-and-pawn code. Pepito 1.59 recognises that a minor up is not enough to win even when accompanied by rooks, though BB vs N is also thought drawish. Two knights is noted as a draw when the opponent has no pawns. There is also code for blind bishops. RESP 0.19 checks for blind bishops, and if a pawnless side is up by a minor or less it is thought drawish unless it is BB vs N or queen versus any two minors. I don’t think two knights is thought drawish. Various recognisers are also used, though I don’t think rook versus a minor is covered. EXchess 5.01beta has code that gives a draw score if the leading side has no less than 3 pieces and no pawns or majors (so KBN/KBB vs K appear to be considered draws). I don’t see any such code in Faile 1.4.

Table 42: Draw recognition, insufficient material

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.6	0.4	0.4	0.3	0.0	0.3	0.5	0.3
RESP	0.6	1.0	0.4	0.4	0.5	0.0	0.3	0.4	0.3
Ryb1	0.4	0.4	1.0	1.0	0.3	0.0	0.3	0.4	0.3
R232	0.4	0.4	1.0	1.0	0.3	0.0	0.3	0.4	0.3
Phal	0.3	0.5	0.3	0.3	1.0	0.0	0.3	0.4	0.3
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.3	0.3	0.3	0.3	0.3	0.0	1.0	0.3	0.2
Pepi	0.5	0.4	0.4	0.4	0.4	0.0	0.3	1.0	0.3
EX5b	0.3	0.3	0.3	0.3	0.3	0.0	0.2	0.3	1.0

2.9 Other

2.9.1 Lazy evaluation

This feature concerns lazy evaluation and how it is used. It is arguably a “search” feature and not one particular to evaluation, but I chose to include it.

Rybka 1.0 Beta has lazy eval (possibly buggy) that depends on the root score at previous iterations, and can also be used based upon a flag in the material table. This is considered before anything else in the evaluation function. Rybka 2.3.2a passes upper/lower bounds to the eval function, and the “positional evaluation” from the previous evaluation is combined with the current material situation. This is considered before everything except the pawn endgame code. Crafty 19.0 keeps track of the maximal amount of post-lazy adjustment (starting at 200cp), and considers lazy eval after special endgame code, passed pawns, trapped bishops, development, and king safety. Pepito 1.59 has a lazy margin which is based on the maximal positional valuation seen, and is checked firstly after pawns, passed pawns, trapped pieces, bishop pair, king security (pawn shelter), seventh rank, and a few other conditions, and secondly after contact squares are additionally done. The loops over pieces come after the second lazy check. RESP 0.19 allows lazy eval after material, bad trades, pawns, and bishops are considered, with additional pre-lazy checks in the endgame. The lazy margins are passed to the eval function. EXchess 5.01beta has a constant lazy margin (EARLY_EXIT) that is applied after pawns, passed pawn evaluation, trapped bishops, and king safety. Faile 1.4, Phalanx XXII and Fruit 2.1 lack lazy evaluation.

Table 43: Lazy evaluation

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.5	0.3	0.4	0.0	0.0	0.0	0.6	0.5
RESP	0.5	1.0	0.3	0.4	0.0	0.0	0.0	0.5	0.5
Ryb1	0.3	0.3	1.0	0.6	0.0	0.0	0.0	0.3	0.3
R232	0.4	0.4	0.6	1.0	0.0	0.0	0.0	0.4	0.3
Phal	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.6	0.5	0.3	0.4	0.0	0.0	0.0	1.0	0.4
EX5b	0.5	0.5	0.3	0.3	0.0	0.0	0.0	0.4	1.0

2.9.2 Material imbalances

This feature involves bonuses for trading or not depending upon the material situation, and more generally material imbalances. Examples include reducing the value of a rook depending on the number of pawns (or vice-versa) with the knight, and any synergy between pieces (such as QN). The bishop pair is considered elsewhere.

Rybka 1.0 and Rybka 2.3.2a have material imbalance tables. To give some indication of these, knights are given a bonus when there are more pawns on the board, rooks are slightly penalised for such, there is a synergy for QN (more than for QB), etc. Crafty 19.0 has a bad trade penalty to avoid giving up a minor for pawns, similarly with a rook for two minors, and a small avoidance of giving up the exchange. There is further code to avoid giving up 3 minors for two rooks or a queen (the penalty is halved for the latter). RESP 0.19 has `bad_trades()` code to try first to avoid giving up the exchange (for pawns or other compensation), secondly to avoid giving up two pieces for a rook, and finally to avoid giving up a minor for pawns. The penalty is the same in all cases. It also has a `QUEEN_DEFENDER` penalty that tries to exchange queens when there is a large material imbalance. Finally, there is a `ROOK_VS_MINOR` bonus for endgames of this sort. Phalanx XXII has a trade-down bonus which inhibits trading when the leading side has no pawns, and otherwise helps the side with more material to favour trades. Faile 1.4 has a small bonus to encourage trading off material when one side has a material advantage, applied after the opening. Pepito 1.59 has a `MAS_PIEZAS` adjustment when one side has an extra piece but not too much extra material. Fruit 2.1 lacks this feature. EXchess 5.01beta has variables for `TRADE_PIECES` and `KEEP_PAWNS`, but neither is used.

Table 44: Material imbalances

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.6	0.3	0.3	0.4	0.3	0.0	0.4	0.0
RESP	0.6	1.0	0.3	0.3	0.4	0.4	0.0	0.4	0.0
Ryb1	0.3	0.3	1.0	1.0	0.3	0.3	0.0	0.3	0.0
R232	0.3	0.3	1.0	1.0	0.3	0.3	0.0	0.3	0.0
Phal	0.4	0.4	0.3	0.3	1.0	0.7	0.0	0.3	0.0
Fail	0.3	0.4	0.3	0.3	0.7	1.0	0.0	0.3	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.4	0.4	0.3	0.3	0.3	0.3	0.0	1.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.9.3 Drawishness from pawn structure

This feature concerns any drawishness considerations from the pawn structure, such as blocked pawns.

Crafty 19.0 has anti-human code that counts the number of “pawn rams” (face-to-face opposing pawns) and penalises according to these. Rybka 2.3.2a considers which of the **b-g** files has pawns of the leading side, and makes a deduction depending on the number of such files and how widespread they are. The other programs do not seem to have this feature.

Table 45: Drawishness from pawn structure

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	1.0	0.0	0.0	0.2	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.2	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
Phal	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.9.4 Hung pieces, pins, and multiple attacks

Phalanx XXII has a computation method involving hung pieces (either attacked pieces that are unguarded, or a lesser piece attacking a stronger one), pinned pieces, and multiple attacks. It is not clear whether this should only be one feature (especially as later engines have used more than once of these), but for now this was my decision.

Table 46: Hung pieces, pins, and multiple attacks

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EX5b	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

2.9.5 Tempo

This is a feature for whether or not the engine gives a bonus for being on move.

Rybka 1.0 Beta gives a bonus of 3 centipawns. Phalanx XXII gives a bonus of 4 centipawns, as does Pepito 1.59, while EXchess 5.01beta has `SIDE_ON_MOVE` at 15. Pepito 1.59 further has `Turno` appear in code for pawn defect imbalance. RESP 0.19, Faile 1.4, Crafty 19.0, Fruit 2.1 and Rybka 2.3.2a lack this feature.

Table 47: Tempo

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RESP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Ryb1	0.0	0.0	1.0	0.0	0.8	0.0	0.0	0.7	0.7
R232	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Phal	0.0	0.0	0.8	0.0	1.0	0.0	0.0	0.8	0.7
Fail	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Fr21	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Pepi	0.0	0.0	0.7	0.0	0.8	0.0	0.0	1.0	0.7
EX5b	0.0	0.0	0.7	0.0	0.7	0.0	0.0	0.7	1.0

3 Accumulation of scores

Table 48: Total Scores

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	36.0	11.8	10.8	10.5	12.9	5.3	11.1	15.2	9.1
RESP	11.8	24.0	8.8	9.2	9.5	5.6	8.9	9.5	10.5
Ryb1	10.8	8.8	32.0	26.0	11.3	5.8	22.7	11.0	8.4
R232	10.5	9.2	26.0	31.0	10.2	5.5	19.3	9.7	7.7
Phal	12.9	9.5	11.3	10.2	42.0	5.9	10.8	16.8	9.6
Fail	5.3	5.6	5.8	5.5	5.9	14.0	5.9	6.2	5.4
Fr21	11.1	8.9	22.7	19.3	10.8	5.9	29.0	10.0	8.0
Pepi	15.2	9.5	11.0	9.7	16.8	6.2	10.0	38.0	11.4
EX5b	9.1	10.5	8.4	7.7	9.6	5.4	8.0	11.4	24.0

Table 49: Evaluation Feature Overlap (Percentage)

-	Craf	RESP	Ryb1	R232	Phal	Fail	Fr21	Pepi	EX5b
Craf	100.0	39.3	31.8	31.3	33.1	21.2	34.2	41.1	30.3
RESP	39.3	100.0	31.4	33.5	28.8	29.5	33.6	30.6	43.8
Ryb1	31.8	31.4	100.0	82.5	30.5	25.2	74.4	31.4	30.0
R232	31.3	33.5	82.5	100.0	27.9	24.4	64.3	28.1	28.0
Phal	33.1	28.8	30.5	27.9	100.0	21.1	30.4	42.0	29.1
Fail	21.2	29.5	25.2	24.4	21.1	100.0	27.4	23.8	28.4
Fr21	34.2	33.6	74.4	64.3	30.4	27.4	100.0	29.9	30.2
Pepi	41.1	30.6	31.4	28.1	42.0	23.8	29.9	100.0	36.8
EX5b	30.3	43.8	30.0	28.0	29.1	28.4	30.2	36.8	100.0

The final graph shows the 30 data points, with the mean of 31.3 that of the 27 “control” points (removing Rybka 2.3.2a), and the shaded regions corresponding roughly to one and two standard deviations ($\sigma \approx 5.6$ for the control group).

