C.04.3.1. Dutch System

Version as agreed by the 83rd FIDE Congress in Istanbul 2012

A Introductory Remarks and Definitions

A.1 Initial ranking list

See C.04.2.B (General Handling Rules - Initial order)

A.2 **Order**

For pairings purposes only, the players are ranked in order of, respectively

a. score
b. pairing numbers assigned to the players accordingly to the initial ranking list and subsequent modifications dependent on possible late entries

A.3 **Score brackets**

Players with equal scores constitute a homogeneous score bracket. Players who remain unpaired after the pairing of a score bracket will be moved down to the next score bracket, which will therefore be heterogeneous. When pairing a heterogeneous score bracket these players moved down are always paired first whenever possible, giving rise to a remainder score bracket which is always treated as a homogeneous one.
A heterogeneous score bracket of which at least half of the players have come from a higher score bracket is also treated as though it was homogeneous.

A.4 **Floats**

By pairing a heterogeneous score bracket, players with unequal scores will be paired. To ensure that this will not happen to the same players again in the next two rounds this is written down on the pairing card. The higher ranked player (called downfloater) receives a downfloat , the lower one (upfloater) an upfloat.

A.5 **Byes**

Should the total number of players be (or become) odd, one player ends up unpaired. This player receives a bye: no opponent, no colour , 1 point or half point (as stated in the tournament regulations).

A.6 **Subgroups - Definition of P0, M0**

a To make the pairing, each score bracket will be divided into two subgroups, to be called S1 and S2, where S2 is equal or bigger than

S1 (for details see C.2 to C.4)
S1 players are tentatively paired with S2 players.

b P0 is the maximum number of pairs that can be produced in each score bracket.
P0 is equal to the number of players divided by two and rounded downwards.

c M0 is the number of players moved down from higher score groups (it may be zero)

A.7 **Colour differences and colour preferences**

The colour difference of a player is the number of games played with white minus the number of games played with black by this player.
After a round the colour preference can be determined for each player who has played at least one game.

a An absolute colour preference occurs when a player’s colour difference is greater than +1 or less than -1, or when a player had the same colour in the two latest rounds he played. The preference is white when the colour difference is less than -1 or when the last two games were played with black. The preference is black when the colour difference is greater than +1, or when the last two games were played with white.

b A strong colour preference occurs when a player‘s colour difference is +1 or -1.
The strong colour preference is white when the colour difference is -1, black otherwise

c A mild colour preference occurs when a player’s colour difference is zero, the preference being to alternate the colour with respect to the previous game.
Before the first round the colour

preference of one player (often the highest one) is determined by lot.

d While pairing an odd-numbered round players having a strong colour preference (players who have had an odd number of games before by any reason) shall be treated like players having an absolute colour preference as long as this does not result in either additionalfloaters or floaters with an higher score or pairs with a higher score difference of the paired players.

e While pairing an even-numbered round players having a mild colour preference (players who have had an even number of games by any reason) shall be treated and counted as if they would have a mild colour preference of that kind (white resp. black) which reduces the number of pairs where both players have the same strong colour preference.

f Players who did not play the first rounds have no colour preference (the preference of their opponents is granted)

A.8 **Definition of X1, Z1**

Provided there are P0 (see A6) pairings possible in a score bracket:

a the minimum number of pairings which must be made in the score bracket, not fulfilling all colour preferences, is represented by the symbol X1.

b in even rounds the minimum number of pairings which must be made in the score bracket, not fulfilling all strong colour preferences (see A7.e), is represented by the symbol Z1

X1 and, in even rounds, Z1 can be

calculated as follows:

w	*in odd rounds: 0; in even rounds: number of players who had an odd number of unplayed games which have a mild colour preference for white (see A7.e)*
b	*in odd rounds: 0; in even rounds: number of players who had an odd number of unplayed games which have a mild colour preference for black (see A7.e)*
W	*(remaining) number of players having a colour preference white*
B	*(remaining) number of players having a colour preference black*
a	*number of players who have not played a round yet*

X1	*If B+b > W+w*	*X1 = P0 – W – w - a,*
	then	*X1 = P0 – B – b - a.*
	else	*X1 = 0*
	If X1 < 0 then	

In even rounds:

Z1	*If B > W then*	*Z1 = P0 - W - b - w - a*
	else	*Z1 = P0 - B - b - w - a.*
	If Z1 < 0 then	*Z1 = 0*

A.9 **Transpositions and exchanges**

a In order to make a sound pairing it is often necessary to change the order in S2. The rules to make such a change, called a transposition, are in D1

b In a homogeneous score bracket it may be necessary to exchange players from S1 to S2. Rules for exchanges are found under D2. After each exchange both S1 and S2 are to be ordered according to A2.

A.10 **Definitions: Top scorers, Backtracking**

Top scorers are players who have a score of over 50% of the maximum possible score when pairing the last round.
Backtracking means to undo the pairings of a higher score bracket to find another set of floaters to the given score bracket.

A.11 **Quality of Pairings - Definition of X and P**

The rules C1 to C14 describe an iteration algorithm to find the best possible pairings within a score bracket.
Starting with the extreme requirement:
P0 pairings with P0 – X1 pairings fulfilling all colour preferences and meeting all requirements B1 to B6
If this target cannot be managed the requirements are reduced step by step to find the best sub-optimal pairings.
The quality of the pairings is defined in descending priority as

- the number of pairs
- the closeness of the scores of the players playing each other
- the number of pairs fulfilling the colour preference of both players (according to A7)
- fulfilling the current criteria for downfloaters
- fulfilling the current criteria for upfloaters

During the algorithm two parameters represent the progress of the iteration:
P is the number of pairings required at a special stage during the pairings algorithm. The first value of P is P0 or M0 and is decreasing.
X is the number of pairings not fulfilling all colour preferences which is acceptable at a special stage during the pairings algorithm. The first value of X is X1 (see A8) and is increasing.

B **Pairing Criteria**

Absolute Criteria
(These may not be violated. If necessary players will be moved down to a lower score bracket.)

B.1 a Two players shall not meet more

than once.

b A player who has received a point or half point without playing, either through a bye or due to an opponent not appearing in time, is a downfloater (see A4) and shall not receive a bye.

B.2 Two players with the same absolute colour preference (see A7.a) shall not meet (therefore no player's colour difference will become >+2 or < -2 nor a player will receive the same colour three times in row)

Note: If it is helpful to reduce the number of floaters or the score of a floater when pairing top scorers B2 may be ignored.
If a top scorer is paired against a non-top scorer, the latter is considered a top scorer for colour allocation purposes.

Relative Criteria
(These are in descending priority. They should be fulfilled as much as possible. To comply with these criteria, transpositions or even exchanges may be applied, but no player should be moved down to a lower score bracket).

B.3 The difference of the scores of two players paired against each other should be as small as possible and ideally zero *(note for programmers: see section D.4 regarding how to use this criterion after repeated application of rule C.13)*

B.4 As many players as possible receive their colour preference

B.5 No player shall receive an identical float in two consecutive rounds.

B.6 No player shall have an identical float as two rounds before.

C **Pairing Procedures**

Starting with the highest score bracket apply the following procedures to all score brackets until an acceptable pairing is obtained. The colour allocation rules (E) are used to determine which players will play with white.

C.1 **Incompatible player**

If the score bracket contains a player for whom no opponent can be found within this score bracket without violating B1 (or B2, except when pairing top scorers) then:

- if this player was moved down from a higher score bracket apply C12.
- if this score bracket is the lowest one apply C13.
- in all other cases: move this player down to the next score bracket

C. **Determine P0, P1, M0, M1, X1, Z1**

2

a Determine P0 according to A6.b. Set P1 = P0
Determine M0 according to A6.c. Set M1= M0

b Determine X1 according to A8.a
In even rounds: determine Z1 according to A8.b

C. 3 **Set requirements P, B2, A7d, X, Z, B5/B6**

a In a homogeneous score bracket set P = P1
In a heterogeneous score bracket set P = M1

b (top scorers) reset B2

c (odd rounds) reset A7.d

d Set X = X1
(even numbered rounds) Set Z = Z1

e (bracket produces downfloaters) reset B5 for downfloaters

f (bracket produces downfloaters) reset B6 for downfloaters

g (heterogeneous score brackets) reset B5 for upfloaters

h (heterogeneous score brackets) reset B6 for upfloaters

C. 4 **Establish sub-groups**

Put the highest P players in S1, all other players in S2.

C. 5 **Order the players in S1 and S2**

According to A2.

C. 6 **Try to find the pairing**

Pair the highest player of S1 against the highest one of S2, the second highest one of S1 against the second highest one of S2, etc. If now P pairings are obtained in compliance with the current requirements the pairing of this score bracket is considered complete.

- in case of a homogeneous or remainder score bracket: remaining players are moved down to the next score bracket. With this score bracket restart at C1.

- in case of a heterogeneous score bracket: only M1 players moved down were paired so far. Mark the current transposition and the value of P (it may be useful later). Redefine P = P1 – M1 Continue at C4 with the remainder group.

C.7 **Transposition**

Apply a new transposition of S2 according to D1 and restart at C6.

C.8 **Exchange**

a In case of a homogeneous (remainder) group: apply a new exchange between S1 and S2 according to D2 and restart at C5.

b In case of a heterogeneous group: if M1 is less than M0, choose another set of M1 players to put in S1 according to D3 and restart at C5

C.9 **Go back to the heterogeneous score bracket (only remainder)**

Terminate the pairing of the homogeneous remainder. Go back to the transposition marked at C6 (in the heterogeneous part of the bracket) and restart from C7 with a new transposition.

C.10 **Lowering requirements**

a (heterogeneous score brackets)
Drop B6 for upfloaters and restart from C.4

b (heterogeneous score brackets)
Drop B5 for upfloaters and restart from C.3.h

c (bracket produces downfloaters)
Drop B6 for downfloaters and restart from C.3.g

d (bracket produces downfloaters)
Drop B5 for downfloaters and restart from C.3.f

e (odd numbered rounds)
If X < P1, increase X by 1 and restart from C.3.e
(even numbered rounds)
If Z < X, increase Z by 1 and restart from C.3.e.
If Z = X and X < P1, increase X by 1, reset Z=Z1 and restart from C.3.e

f (odd numbered rounds)
Drop A7.d and restart from C.3.d

g (top scorers)

Drop B2 and restart from C.3.c

Any criterion may be dropped only for the minimum number of pairs in the score bracket

C.11 **Deleted**

(see C.10.e)

C.12 **Backtrack to previous Score bracket**

If there are moved down players: backtrack to the previous score bracket. If in this previous score bracket a pairing can be made whereby another set of players of the same size and with the same scoreswill be moved down to the current one, and this now allows P1 pairings to be made then this pairing in the previous score bracket will be accepted.
Backtracking is disallowed when already backtracking from a lower score bracket

C.13 **Lowest Score Bracket**

In case of the lowest score bracket: if it is heterogeneous, try to reduce the number of pairable moved-down players (M1), as shown in C14.b2. Otherwise backtrack to the penultimate score bracket. Try to find another pairing in the penultimate score bracket which will allow a pairing in the lowest score bracket. If in the penultimate score bracket P becomes zero (i.e. no pairing can be found which will allow a correct pairing for the lowest score bracket) then the two lowest score brackets are joined into a new lowest score bracket. Because now another score bracket is the penultimate one, C13 can be repeated until an acceptable pairing is obtained.
Such a merged score bracket shall be treated as a heterogeneous score bracket with the latest added score bracket as S1.

C.14 **Decrease P1, X1, Z1, M1**

a **For homogeneous score brackets**:
As long as P1 is greater than zero, decrease P1 by 1.
If P1 equals zero the entire score bracket is moved down to the next one. Start with this score bracket at C1
Otherwise, as long as X1 is greater than zero, decrease X1 by 1.
In even rounds, as long as Z1 is greater than zero, decrease Z1 by 1.
Restart from C3.a

b **For heterogeneous score brackets:**

1 If the pairing procedure has got to the remainder at least once, reduce P1, X1 and, in even rounds, Z1 as in the homogeneous score brackets

and restart from C3.a

2 Otherwise, as long as M1 is greater than 1, reduce M1 by 1 and restart from C3.a
If M1 is one, set M1=0, manage the bracket as homogeneous, set P1=P0 and restart from C2.b.

D **Transposition and exchange procedures**

D.1 **Transpositions**

D1.1 **Homogeneous or remainder score brackets**

Example: S1 contains 5 players 1, 2, 3, 4, 5 (in this sequence)
S2 contains 6 players 6, 7, 8, 9, 10, 11 (in this sequence)

Transpositions within S2 should start with the lowest player, with descending priority

0. 6 – 7 – 8 – 9 – 10 – 11
1. 6 – 7 – 8 – 9 – 11 – 10
2. 6 – 7 – 8 – 10 – 9 – 11
3. 6 – 7 – 8 – 10 – 11 – 9
4. 6 – 7 - 8 – 11 – 9 – 10
5. 6 – 7 – 8 – 11 – 10 – 9
6. 6 – 7 – 9 – 8 – 10 – 11
7. 6 – 7 – 9 – 8 – 11 – 10
8. 6 – 7 – 9 – 10 – 8 – 11
9. 6 – 7 – 9 – 10 – 11 – 8
10. 6 – 7 – 9 – 11 – 8 – 10

11. 6 – 7 – 9 – 11 – 10 – 8
12. 6 – 7 – 10 – 8 – 9 – 11
13. 6 – 7 – 10 – 8 – 11 – 9
14. 6 – 7 – 10 – 9 – 8 – 11
15. 6 – 7 – 10 – 9 – 11 – 8
16. 6 – 7 – 10 – 11 – 8 – 9
17. 6 – 7 – 10 – 11 – 9 – 8
18. 6 – 7 – 11 – 8 – 9 – 10
19. 6 – 7 – 11 – 8 – 10 – 9
20. 6 – 7 – 11 – 9 – 8 – 10
21. 6 – 7 – 11 – 9 – 10 – 8
22: 6 – 7 – 11 – 10 – 8 – 9
23. 6 – 7 – 11 – 10 – 9 – 8
24. 6 – 8 – 7 - …..

To be continued. (at all 720 figures)

719. 11 – 10 – 9 – 8 – 7 – 6

D1.2 **Heterogeneous score brackets**

The algorithm is in principle the same as for homogeneous score brackets (See D1.1), especially when S1 = S2, If S1 < S2 the algorithm must be adapted to the difference of players in S1 and S2.

Example: S1 contains 2 players 1, 2, (in this sequence)
S2 contains 6 players 3, 4, 5, 6, 7, 8 (in this sequence)

The transpositions within S2 are the same as in D1.1. But only the S1 first listed players of a transposition may be paired with S1. The other S2 – S1 players remain unpaired in this attempt.

D.2 **Exchange of players (homogeneous or remainder score bracket only)**

When applying an exchange between S1 and S2 the difference between the numbers exchanged should be as small as possible. When differences of various options are equal take the one concerning the lowest player of S1. Then take the one concerning the highest player of S2.

General procedure:

- Sort the groups of players of S1 which may be exchanged in decreasing lexicographic order as shown below in the examples (List of S1 exchanges)
- Sort the groups of players of S2 which may be exchanged in increasing lexicographic order as shown below in the examples (List of S2 exchanges)
- The difference of numbers of players concerned in an exchange is: (Sum of numbers of players in S2) – (Sum of numbers of players in S1). This difference shall be as small as possible.
- When differences of various options are equal:
  - Take at first the option top down from the list of S1 exchanges.
  - Take then the option top down from the list of S2 exchanges.
- After each exchange both S1 and S2 should be ordered according to A2

Remark: Following this procedure it may occur that pairings already checked will appear again. These repetitions are harmless because they give no better pairings than at their first occurrence.

Example for the exchange of one player:

		S1				
		5	**4**	**3**	**2**	**1**
S2	**6**	1	3	6	10	15
	7	2	5	9	14	20
	8	4	8	13	19	24
	9	7	12	18	23	27

	10	11	17	22	26	29
	11	16	21	25	28	30

1. exchange player 5 from S1 with player 6 from S2 : difference 1
2. exchange player 5 from S1 with player 7 from S2 : difference 2
3. exchange player 4 from S1 with player 6 from S2 : difference 2

Etc.

Example for the exchange of two players:

		S1									
		5,4	**5,3**	**5,2**	**5,1**	**4,3**	**4,2**	**4,1**	**3,2**	**3,1**	**2,1**
S2	**6,7**	1	3	7	14	8	16	28	29	45	65
	6,8	2	6	13	24	15	27	43	44	64	85
	6,9	4	11	22	37	25	41	60	62	83	104
	6,10	9	20	35	53	39	58	79	81	102	120
	6,11	17	32	50	71	55	76	96	99	117	132
	7,8	5	12	23	38	26	42	61	63	84	105
	7,9	10	21	36	54	40	59	80	82	103	121
	7,10	18	33	51	72	56	77	97	100	118	133
	7,11	30	48	69	90	74	94	113	115	130	141
	8,9	19	34	52	73	57	78	98	101	119	134
	8,10	31	49	70	91	75	95	114	116	131	142
	8,11	46	67	88	108	92	111	126	128	139	146
	9,10	47	68	89	109	93	112	127	129	140	147
	9,11	66	87	107	123	110	125	137	138	145	149

	10,11	86	106	122	135	124	136	143	144	148	150

1. Exchange 5,4 from S1 with 6,7 from S2: difference = 4
2. Exchange 5,4 from S1 with 6,8 from S2: difference = 5
3. Exchange 5,3 from S1 with 6,7 from S2: difference = 5
4. Exchange 5,4 from S1 with 6,9 from S2: difference = 6
5. Exchange 5,4 from S1 with 7,8 from S2: difference = 6
6. Exchange 5,3 from S1 with 6,8 from S2: difference = 6

Etc.

Example for the exchange of three players:
List of S1 exchanges:
5,4,3 5,4,2 5,4,1 5,3,2 5,3,1
5,2,1 4,3,2 4,3,1 4,2,1 3,2,1
List of S2 exchanges:
6,7,8 6,7,9 6,7,10 6,7,11 6,8,9 6,8,10
6,8,11 6,9,10 6,9,11 6,10,11 7,8,9 7,8,10
7,8,11 7,9,10 7,9,11 7,10,11 8,9,10 8,9,11
8,10,11 9,10,11

1. Exchange 5,4,3 from S1 with 6,7,8 from S2: difference = 9
2. Exchange 5,4,3 from S1 with 6,7,9 from S2: difference = 10
3. Exchange 5,4,2 from S1 with 6,7,8 from S2: difference = 10
4. Exchange 5,4,3 from S1 with 6,7,10 from S2: difference = 11
5. Exchange 5,4,3 from S1 with 6,8,9 from S2: difference = 11
6. Exchange 5,4,2 from S1 with 6,7,9 from S2: difference = 11

Etc.

Exact procedure for exchange of N (N= 1,2,3,4..) players in a scoregroup of P players

- Sort all possible subsets of N players of S1 in decreasing lexicographic order to an array S1LIST which may have S1NLIST elements.
- Sort all possible subsets of N players of S2 in increasing lexicographic order to an array S2LIST which may have S2NLIST elements
- To each possible exchange between S1 and S2 can be assigned a difference which is a number defined as:

(Sum of numbers of players in S2, included in that exchange) - (Sum of numbers of players in S2, included in that exchange)

In functional terms:
DIFFERENZ(I,J) = sum of numbers of players of S2 in subset J – sum of numbers of players of S1 in subset I

This difference has a minimum DIFFMIN = DIFFERENZ (1,1)
and a maximum DIFFMAX = DIFFERENZ (S1NLIST, S2NLIST)

Now the procedure to find the exchanges in correct order:

```
1    DELTA = DIFFMIN
2    I=1 J=1
3    If DELTA = DIFFERENZ(I,J) then do this exchange, after  that goto 4
4    if J < S2NLIST then J=J+1 goto 3
5    if I<S1NLIST  then  I=I+1, J=1 goto 3
6    DELTA =DELTA+1
7    if DELTA > DIFFMAX  goto 9
8    goto 2
9    The possibilities to exchange N players are exhausted
```

After each exchange both S1 and S2 should be ordered according to A2

D.3 **Moved-down players exchange**

Example: M0 is 5; The players originally in S1 are {1, 2, 3, 4, 5}

The elements in S1 start with the M1 highest players, then with descending priority:

	S1 elements in descending priority				
	M1 = 5	**M1 = 4**	**M1 = 3**	**M1 = 2**	**M1 = 1**
	1-2-3-4-5	1-2-3-4	1-2-3	1-2	1
		1-2-3-5	1-2-4	1-3	2
		1-2-4-5	1-2-5	1-4	3
		1-3-4-5	1-3-4	1-5	4
M0 = 5		2-3-4-5	1-3-5	2-3	5
			1-4-5	2-4	
			2-3-4	2-5	
			2-3-5	3-4	
			2-4-5	3-5	
			3-4-5	4-5	

D.4 **Note for programmers: B.3-factor in the lowest score bracket**

After repeated applications of rule C13, it is possible that the lowest score bracket (LSB)

contains players with many different scores and that there are multiple ways to pair them. Such a bracket either is homogeneous (when the number of players coming from the penultimate score bracket is equal or higher than the number of LSB players) or eventually produces a homogeneous remainder.
The following rule must be followed by pairing programs: **The best pairing for such a homogeneous score bracket or remainder is the one that minimizes the sum of the squared differences between the scores of the two players in each pair *(called B3-factor)*. Getting the bye is equivalent to face an opponent with one point less than the lowest ranked player (even if this is resulting in -1).**
Example: Let the following be the players in the LSB:
3.0 : A
2.5 : B, C
2.0 : D
1.5 : E
1.0 : F

F can only play against A.

The pairing will initially start with S1={A,B,C} S2={D,E,F} and, after a few transpositions, it will move to **Png1: [S1={A,B,C} S2={F,D,E}]**. Work is not finished, though. Some exchanges must be applied to get to **Png2: [S1={A,B,D} S2={F,C,E}]** which is the best possible pairing. This is because of the B3-factor. Let us compute it:

Png1: (A-F, B-D, C-E) => (2.0*2.0 + 0.5*0.5 + 1.0*1.0) = 5.25
Png2: (A-F, B-C, D-E) => (2.0*2.0 + 0.0*0.0 + 0.5*0.5) = 4.25
Warning: if there is a seventh player (G) with less than 2.5 points, who is the only one who can get the bye, the LSB is heterogeneous and no exchanges in S1 are allowed. In such an instance, the pairing of the LSB is: A-F, B-D, C-E, G(bye)
Remark: This algorithm is nothing especial. It is the best mathematical method to find the pairings which an arbiter seeing all the player's data naturally will achieve.

E Colour Allocation rules

For each pairing apply (with descending priority):

E.1 Grant both colour preferences

E.2 Grant the stronger colour preference

E.3 Alternate the colours to the most recent round in which they played with different colours

E.4 Grant the colour preference of the higher ranked player

E.5 In the first round all even numbered players in S1 will receive a colour different from all odd numbered players in S1