

Abstract :

We modify the present rule of soccer in order to be able to get points of plural kinds referring to the point system of rugby.

1. Tandem kick

If CK, DFK occur, doing before them of LK, MK, NK of which points are smaller than usual goal and LK, MK, NK in this case are called tandem kick. In the last, LK, MK, NK are inserted selectively, however, tandem kick does not have a clear influence on a game. If success or failure of a goal is expressed on LK, MK, NK, a gap takes place like conversion goal in rugby. Therefore, we can insert LK, MK, NK before CK, DFK in series not having a clear influence on a game. In tandem kick, game clock is paused and its points are not reflected in score.

Figure 1 is CK's case. If referee signals CK and a ball is placed on corner area, game clock is paused. If referee resignals CK and whistles after LK, MK, game clock is resumed.

Figure 2 is DFK's case. If referee signals DFK and a ball is placed on fixed point, game clock is paused. If a ball is placed on fixed point, referee resignals DFK and whistles after NK, game clock is resumed.

In the figures, game clock is paused and resumed. Referee may make the time required between signal and resignal additional time if it is not a burden to referee. If the time required is made constant, it is called constant insertion time. If we use constant insertion time(for example, minute size=0.5, 1), the summation work on additional time is not needed.

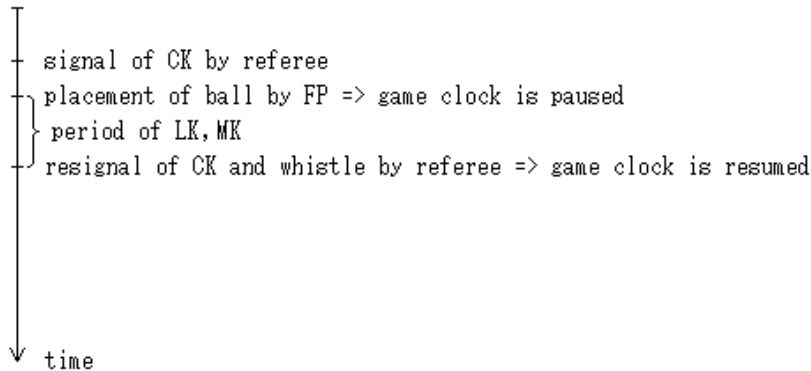


Figure 1

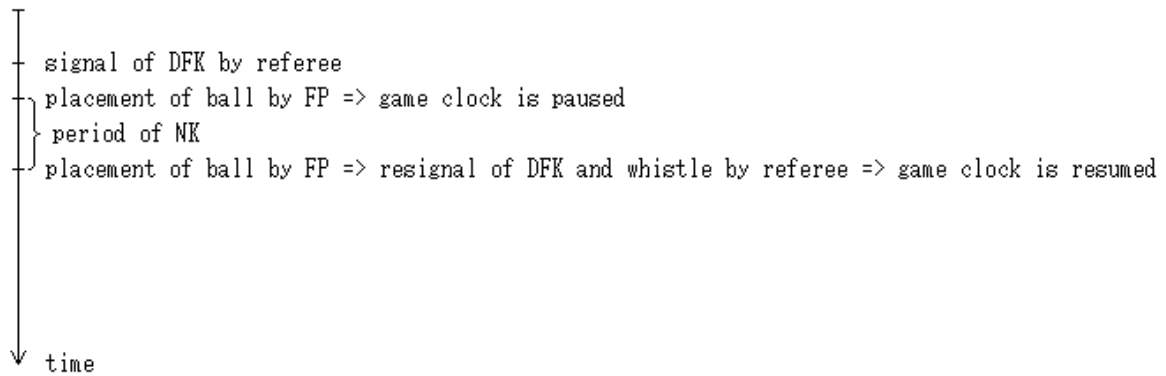


Figure 2

2. E league

We consider experimental score besides game score. Experimental score is made from points of tandem kick and game score (number of goals). Points of tandem kick (LK, MK, NK) are according to the last, however, there is a possibility of modification of the number of usual goals.

A probability of each goal is got from the number of goals of tandem kick. Experimental score is number of usual goals $\times 7$ + total points of LK, MK, NK. If a goal is got after CK, DFK the ball not crossing centre line, the ball not being caught by GK and a goal kick is not done, the goal is not counted in experimental score. To tell it inversely, if the ball crosses centre line, the ball is caught by GK or a goal kick is done, a goal after it is counted. Figure 3 shows this situation. Because it is impossible that a full set of Fine soccer is done in parallel, we try getting an approximation of score in Fine soccer namely experimental score with a subset.

If experimental score is got, we get processed score from this. In real Fine soccer, we do not do LK, MK, NK always if CK, DFK occur. So, we get a combination between the number of LKs, the number of MKs and the number of NKs for score to exceed opponent's score or become maximum assuming the number of LKs, the number of MKs and the number of NKs to be variables theoretically. We call it processed score. In this case, opponent's score is also processed score which is got from experimental score. In processed score, if we select not LK, MK, NK but CK, DFK, the uncounted goals mentioned earlier revive.

We call a league game in which tandem kick is done E league. In E league, staffs, managers and players must understand meaning and rules of it and it is also needed that spectators are explained meaning to and understand it.

Because we cover combinations of "alternatives" with more than one E league, we rename E league E_j league ($j = 1, 2, 3, \dots$) in order to distinguish combinations of "alternatives". The following are "alternatives" in E_j . The figure numbers are the last figure numbers.

- E_1 : Fig.1 left, Fig.3, Fig.5, on goal line
- E_2 : Fig.1 left, Fig.3, Fig.5, in goal area
- E_3 : Fig.1 left, Fig.2, Fig.4, on goal line
- E_4 : Fig.1 left, Fig.2, Fig.4, in goal area
- E_5 : Fig.1 right, Fig.3, Fig.5, on goal line
- E_6 : Fig.1 right, Fig.3, Fig.5, in goal area
- E_7 : Fig.1 right, Fig.2, Fig.4, on goal line
- E_8 : Fig.1 right, Fig.2, Fig.4, in goal area

In $E_5 \sim E_8$, on CK from FP, tandem kick is done only in the case that the ball crosses semiline touching FP.

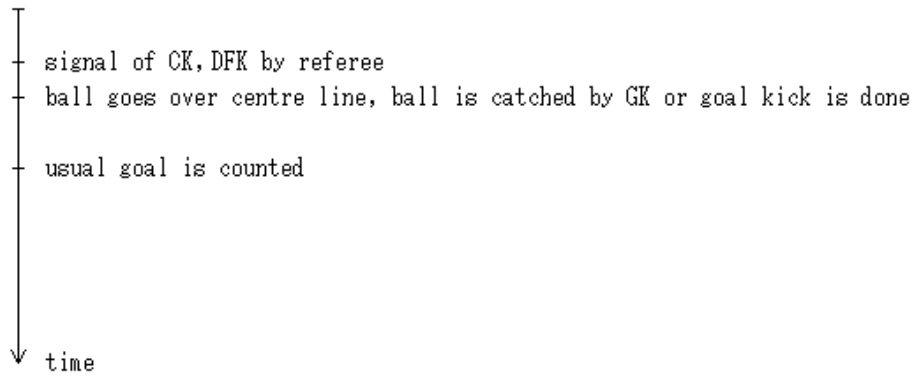


Figure 3

3. Decision points(1)

Tandem Kick can be also used as a draw measure which lessens a rate of penalty shoot-out. It is done over first and second half(90 minutes) and we decide a game with its points if two scores are the same after second half. Tandem Kick is done only when CK occurs in principle. "alternatives" and points of LK, MK are set like the following:

- "alternatives" : the same as E_1 league
- LK : 7 points
- MK : 2 points

We get a draw rate on tandem kick over first and second half. Parameters in the last program are modified like the following:

```
#define lmd1a (1.8)
#define lmd1b (0)
#define lmd1c (3.5)
#define lmd1d (0)

#define lmd2a (1.8)
#define lmd2b (0)
#define lmd2c (3.5)
#define lmd2d (0)

#define ra 7
#define rb 0
#define rc 2
#define rd 0
```

From this

- rate of draw : 0.034844
- mean score : 19.599916
- modal score : 13 (rate=0.064204)

- variance : 102.195310

Because a draw rate on first and second half is 0.179749, the draw rate in consideration of the above rate becomes $0.179749 \times 0.034844 = 0.006263$.

In a tournament game, extended game(30 minutes) is possible. We decide a game with tandem kick points in extended game if two scores are the same after second half of extended game. In tandem kick in extended game

```
#define lmd1a ((1.8/90.)*30)
#define lmd1b (0)
#define lmd1c ((3.5/90.)*30)
#define lmd1d (0)
```

```
#define lmd2a ((1.8/90.)*30)
#define lmd2b (0)
#define lmd2c ((3.5/90.)*30)
#define lmd2d (0)
```

```
#define ra 7
#define rb 0
#define rc 2
#define rd 0
```

From this

- rate of draw : 0.118083
- mean score : 6.533297
- modal score : 2 (rate=0.199385)
- variance : 34.065051

Because a draw rate on extended game is 0.336513, the draw rate in consideration of the above rate becomes $0.336513 \times 0.118083 = 0.039736$. If 90 minutes + 30 minutes, it becomes $0.006263 \times 0.039736 = 0.000249$. If tandem kick points in extended game are the same, we decide a game with penalty shoot-out or drawing lots.

In part of sports, a game is decided with decision if the game is not decided within regular time. Decision points represents a degree of attack and it is given according to precision and quality of a shoot in principle in Fine soccer. This is tandem kick points.

4. Decision points(2)

We can use ball point and excluded zone in PK as exceptions of "alternatives" in tandem kick.

- "alternatives" : Fig.1 left, penalty mark, Fig.4, on goal line
- LK : 7 points
- MK : 2 points

In this case, the difference between LK and MK is only points. Because $lmd1a, lmd2a = 3 \times 0.8$, $lmd1c, lmd2c = 7 \times 0.8$ assuming the probability of goal in PK to be 0.8

```
#define lmd1a (2.4)
#define lmd1b (0)
#define lmd1c (5.6)
#define lmd1d (0)
```

```
#define lmd2a (2.4)
#define lmd2b (0)
#define lmd2c (5.6)
#define lmd2d (0)
```

```
#define ra 7
#define rb 0
#define rc 2
#define rd 0
```

From this

- rate of draw : 0.025745
- mean score : 27.999898
- modal score : 24 (rate=0.045006)
- variance : 139.994068

Because a draw rate on first and second half is 0.179749, the draw rate in consideration of the above rate becomes $0.179749 \times 0.025745 = 0.004628$.

In tandem kick in extended game

```
#define lmd1a ((2.4/90.)*30)
#define lmd1b (0)
#define lmd1c ((5.6/90.)*30)
#define lmd1d (0)
```

```
#define lmd2a ((2.4/90.)*30)
#define lmd2b (0)
#define lmd2c ((5.6/90.)*30)
#define lmd2d (0)
```

```
#define ra 7
#define rb 0
#define rc 2
#define rd 0
```

From this

- rate of draw : 0.076048
- mean score : 9.333272
- modal score : 2 (rate=0.129702)
- variance : 46.663849

Because a draw rate on extended game is 0.336513, the draw rate in consideration of the above rate becomes $0.336513 \times 0.076048 = 0.025591$. If 90 minutes + 30 minutes, it becomes $0.004628 \times 0.025591 = 0.000118$.

5. Points

In the last, points of goal in LK, MK, NK are "alternatives", however, because we must get a probability of each goal in order to fix these points finally, the result of E_j league or Decision points(1) is needed. In the last, it is described that we set MK as a score by for example, 95% point of a statistical distribution of MK goals becomes less than 7 points of usual goal. First, we get the mean of MK goals $lmd1c$ from the mean number of MKs per game \times the probability. Second, we get 95% point from the statistical distribution of this $lmd1c$. It is an ideal that this 95% point multiplied by points becomes less than 7 points. Therefore, we need to get the probability first thing. So, points in E_j league are set temporarily like the following:

```
#define ra 7
#define rb 3
#define rc 1
#define rd 2
```

6. Gathering of tandem kicks

In tandem kick, because LK, MK, NK which do not have an influence on a score during a game are inserted, time management is needed. If it is troublesome and dampens stream of game, we may do LK, MK not in a game but together after first and second half(90 minutes), extended game(30 minutes). However, LK, MK are not done if a game is decided in the case that we get decision points.

7. Break

I think that a break may be inserted if success of an usual goal is expressed and success or failure of a goal is expressed on LK, MK, NK. It is inserted on the former case in usual soccer and on the both cases in Fine soccer.

I think that in rugby, a break may be inserted if success or failure of a conversion goal is expressed and success of a penalty goal, a drop goal is expressed.

We can make a demand for constant time-out during break. The size of the time-out is made to the extent of the size of handball's time-out. In this case, the time which is added to additional time or each half becomes break + time-out.

フラインサッカー (2)

菊池盛雄

アブストラクト :

ラグビーの得点体系を参考にして複数の得点が入るように現在のサッカーのルールを修正します。

1. タンデムキック

CK、DFKが発生したらその前に通常のゴールよりも小さい得点が入るLK、MK、NKを行うこと、およびこの場合のLK、MK、NKをタンデムキックと称します。前回においてはLK、MK、NKは選択的に挿入されましたが、タンデムキックは試合に明白な影響を与えません。LK、MK、NKに関してゴールの成否が示されればラグビーのコンバージョンゴールと同様ギャップが生じます。したがって、LK、MK、NKはCK、DFKの前に試合に明白な影響を与えることなく直列に挿入することが可能なのです。タンデムキックにおいては時計は pause され、その得点はスコアに反映されません。

図1はCKの場合です。レフェリーがCKを signal し、ボールがコーナーエリアに置かれると時計は pause されます。LK、MKが終了してレフェリーがCKを resignal し、whistle したら時計は resume されます。

図2はDFKの場合です。レフェリーがDFKを signal し、ボールが所定の位置に置かれると時計は pause されます。NKが終了してボールが所定の位置に置かれ、レフェリーがDFKを resignal し、whistle したら時計は resume されます。

図では game clock が pause、resume されますが、過負担でなければ signal と resignal の間の所要時間をレフェリーが additional time 化してもかまいません。この所要時間を一定とした場合、これを定挿入時間と称します。定挿入時間 (たとえば 0.5、1 分) を用いればこの所要時間を毎回 additional time 化する必要はありません。

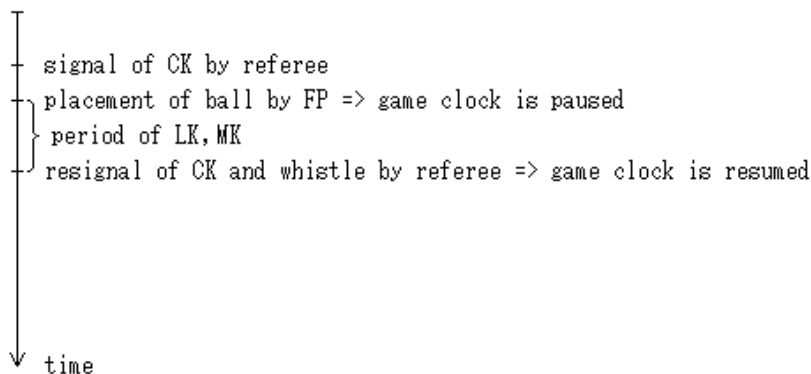


図 1

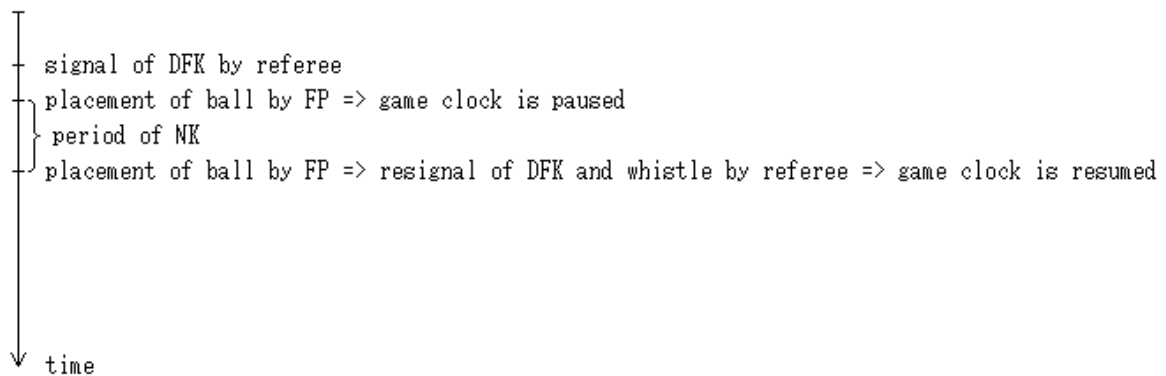


図 2

2. E リーグ

試合のスコアの他に実験スコアというものを考えましょう。実験スコアはタンデムキックの得点と試合のスコア(ゴール数)から作られます。タンデムキック(LK、MK、NK)の得点は前回の通りですが、通常のゴール数が修正される可能性があります。

タンデムキックのゴール数から各々のゴールの確率が求められます。実験スコアは、通常のゴール数 $\times 7 + \text{LK、MK、NK}$ の総得点です。もしCK、DFKの後でボールがセンターラインを越えることなく、かつボールがGKにキャッチされることなく、かつゴールキックがなされることなくゴールが成れば、実験スコアではこのゴールはカウントされません。逆に、もしボールがセンターラインを越えるか、またはボールがGKにキャッチされるか、またはゴールキックがなされれば、その後のゴールはカウントされます。図3はこの事情を表しています。フラインサッカーのフルセットを並行して行うことは不可能なので、サブセットでフラインサッカーのスコアの近似値すなわち実験スコアを得ようという訳です。

実験スコアが得られたらこれから加工スコアを求めます。実際のフラインサッカーではCK、DFKが発生したらすべてにおいてLK、MK、NKを行う訳ではありません。そこで、机上でLK、MK、NKの数を変数としてスコアが相手チームのスコアより大きくなる、または最大になるようなLK、MK、NKの数の組合せを求めます。このスコアを加工スコアと称します。この場合、相手チームのスコアも実験スコアから求められる加工スコアです。加工スコアにおいて、LK、MK、NKではなくCK、DFKを選択した場合は、先に述べたカウントされないゴールは復活します。

タンデムキックを行うリーグ戦をEリーグと称します。Eリーグでは、スタッフ、監督、選手がEリーグの趣旨、ルールを理解していることが必要ですが、観客にも趣旨を説明して理解してもらわなければなりません。

複数のEリーグで”alternatives”の組合せを網羅するので、”alternatives”の組合せを区別するためにEリーグを E_j リーグ($j = 1, 2, 3, \dots$)とリネームします。以下は E_j の”alternatives”です。図番号は前回の図番号です。

- E_1 : 図1左、図3、図5、ゴールライン上
- E_2 : 図1左、図3、図5、ゴールエリア内
- E_3 : 図1左、図2、図4、ゴールライン上
- E_4 : 図1左、図2、図4、ゴールエリア内
- E_5 : 図1右、図3、図5、ゴールライン上
- E_6 : 図1右、図3、図5、ゴールエリア内
- E_7 : 図1右、図2、図4、ゴールライン上
- E_8 : 図1右、図2、図4、ゴールエリア内

E_5 から E_8 では、FP起因のCKに関しては、ボールがFPに触れてセミラインを越えた場合だけタンデ

ムキックを行います。

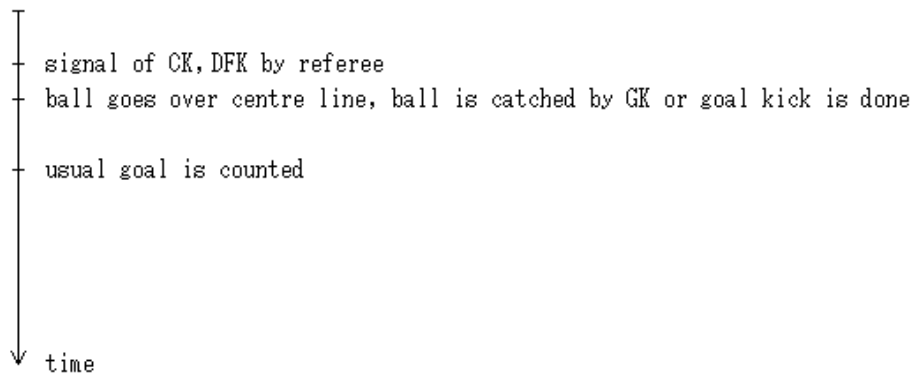


図 3

3. 判定点 (1)

タンデムキックは PK 戦の割合を少なくする引き分け対策としても使えます。前後半 90 分においてタンデムキックを行い、もし後半終了時同点である場合はこの得点で勝敗を決めます。タンデムキックは原則 CK が発生した場合に行います。”alternatives”と LK、MK の得点は以下のように設定します。

- ・”alternatives”：E₁ リーグと同じ
- ・LK：7 点
- ・MK：2 点

前後半におけるタンデムキックの引き分けの割合を求めましょう。前回のプログラムにおいてパラメーターを以下のように修正します。

```
#define lmd1a (1.8)
#define lmd1b (0)
#define lmd1c (3.5)
#define lmd1d (0)

#define lmd2a (1.8)
#define lmd2b (0)
#define lmd2c (3.5)
#define lmd2d (0)

#define ra 7
#define rb 0
#define rc 2
#define rd 0
```

これから

- ・引き分けの割合：0.034844
- ・平均スコア：19.599916
- ・最頻スコア：13 (割合=0.064204)
- ・分散：102.195310

前後半の引き分けの割合は 0.179749 なので、上述の割合を考慮した引き分けの割合は、 $0.179749 \times 0.034844 = 0.006263$ となります。

トーナメント戦では延長戦 30 分があり得ます。もし延長戦後半終了時同点である場合は、延長戦におけるタンデムキックの得点で勝敗を決めます。延長戦におけるタンデムキックでは

```
#define lmd1a ((1.8/90.)*30)
#define lmd1b (0)
#define lmd1c ((3.5/90.)*30)
#define lmd1d (0)

#define lmd2a ((1.8/90.)*30)
#define lmd2b (0)
#define lmd2c ((3.5/90.)*30)
#define lmd2d (0)

#define ra 7
#define rb 0
#define rc 2
#define rd 0
```

これから

- ・引き分けの割合：0.118083
- ・平均スコア：6.533297
- ・最頻スコア：2 (割合=0.199385)
- ・分散：34.065051

延長戦の引き分けの割合は 0.336513 なので、上述の割合を考慮した引き分けの割合は、 $0.336513 \times 0.118083 = 0.039736$ となります。90 分 + 30 分では $0.006263 \times 0.039736 = 0.000249$ となります。延長戦におけるタンデムキックの得点と同じなら PK 戦または抽選で勝敗を決めます。

一部のスポーツでは規定時間内に勝敗が決まらなると勝敗が判定で決められます。判定点は試合中の攻撃の程度を表すものであり、フラインサッカーでは原則シュートの精度と質に応じて与えられます。これがタンデムキックの得点なのです。

4. 判定点 (2)

タンデムキックの”alternatives”の例外として、PK におけるボールの位置、排除ゾーンを用いることができます。

- ・”alternatives”：図 1 左、ペナルティマーク、図 4、ゴールライン上
- ・LK：7 点
- ・MK：2 点

この場合、LK と MK の違いは得点だけです。PK におけるゴールの確率を 0.8 とすれば $lmd1a$ 、 $lmd2a = 3 \times 0.8$ 、 $lmd1c$ 、 $lmd2c = 7 \times 0.8$ であるので

```
#define lmd1a (2.4)
#define lmd1b (0)
```

```
#define lmd1c (5.6)
```

```
#define lmd1d (0)
```

```
#define lmd2a (2.4)
```

```
#define lmd2b (0)
```

```
#define lmd2c (5.6)
```

```
#define lmd2d (0)
```

```
#define ra 7
```

```
#define rb 0
```

```
#define rc 2
```

```
#define rd 0
```

これから

- ・引き分けの割合：0.025745
- ・平均スコア：27.999898
- ・最頻スコア：24 (割合=0.045006)
- ・分散：139.994068

前後半の引き分けの割合は0.179749なので、上述の割合を考慮した引き分けの割合は、 $0.179749 \times 0.025745=0.004628$ となります。

延長戦におけるタンデムキックでは

```
#define lmd1a ((2.4/90.)*30)
```

```
#define lmd1b (0)
```

```
#define lmd1c ((5.6/90.)*30)
```

```
#define lmd1d (0)
```

```
#define lmd2a ((2.4/90.)*30)
```

```
#define lmd2b (0)
```

```
#define lmd2c ((5.6/90.)*30)
```

```
#define lmd2d (0)
```

```
#define ra 7
```

```
#define rb 0
```

```
#define rc 2
```

```
#define rd 0
```

これから

- ・引き分けの割合：0.076048
- ・平均スコア：9.333272
- ・最頻スコア：2 (割合=0.129702)
- ・分散：46.663849

延長戦の引き分けの割合は0.336513なので、上述の割合を考慮した引き分けの割合は、 $0.336513 \times 0.076048=0.025591$ となります。90分+30分では $0.004628 \times 0.025591=0.000118$ となります。

5. 得点

前回においてはLK、MK、NKによるゴールの得点は”alternatives”となっていますが、これらの得点を最終的に定めるためには各々のゴールの確率を求めなければならないので、 E_j リーグまたは判定点(1)の結果が必要となります。前回において、MKは、MKによるゴールの統計学的分布のたとえば95%点による得点が通常のゴールの7点未満になる、というように設定します、と述べました。まず1試合当たりのMKの平均回数×確率からMKによるゴールの平均 $lmd1c$ を求めます。次にこの $lmd1c$ の統計学的分布から95%点を求めます。この95%点に得点をかけたものが7点未満になればよい訳です。したがって、先に確率を求める必要があるのです。そこで、 E_j リーグにおける得点は暫定的に以下のように設定します。

```
#define ra 7
#define rb 3
#define rc 1
#define rd 2
```

6. タンデムキックの集約

タンデムキックにおいては試合中にスコアに影響を及ぼさないLK、MK、NKを挿入するので時間管理が必要です。タンデムキックが煩らしくて試合の流れをそくようなことがあれば、LK、MKは試合中ではなく前後半90分、延長戦30分が終了してからまとめて行ってもかまいません。ただし、判定点を求める場合において勝敗が決まっていればLK、MKは行いません。

7. ブレイク

通常のゴールの成功が示されれば、およびLK、MK、NKに関してゴールの成否が示されればブレイクをさしはさんでもよいと思います。通常のサッカーでは前者に関して、フラインサッカーでは両者に関してです。

ラグビーにおいては、コンバージョンゴールの成否が示されれば、およびペナルティゴール、ドロップゴールの成功が示されればブレイクをさしはさんでもよいと思います。

ブレイクの間にレフェリーに一定のタイムアウトを要求できます。このタイムアウトのサイズはハンドボールのそれ程度とします。この場合、additional timeまたは各ハーフに付加される時間はブレイク+タイムアウトとなります。

List 1:markov.c

```
/* markov.c */
/* by Morio Kikuchi 2017.1.1 */
/* COMPILER:gpp(dos) */
/* COMMANDLINE:gcc -Dfar= -o draw.exe draw.c */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include <malloc.h>

#define LW 0 /* 0,1 */
/*#define Rugby*/
/*#define Multi*/
#define TCTCS 0 /* 0,1,2 for LW=1 */

#ifndef Rugby
/* Soccer */
#ifndef Multi
#define tsum 90.
#define mnt 90
#define dmax 10
#define R1a 0.015
#define R1a_ 0.03
#define R2a 0.012
#define R2a_ 0.04
#else
#define tsum 40.
#define mnt 40
#define dmax 100
#define R1a 0.2
#define R1b 0.5
#define R1c 0.2
#define R1a_ 0
#define R1b_ 0
#define R1c_ 0
#define R2a 0.2
#define R2b 0.45
#define R2c 0.15
#define R2a_ 0
#define R2b_ 0
#define R2c_ 0
```

```

#define ra 3
#define rb 2
#define rc 1

/*#define tsum 90.
#define mnt 90
#define dmax 10
#define R1a (2.4/tsum)
#define R1b (0/tsum)
#define R1c (5.6/tsum)
#define R1a_ 0
#define R1b_ 0
#define R1c_ 0
#define R2a (2.4/tsum)
#define R2b (0/tsum)
#define R2c (5.6/tsum)
#define R2a_ 0
#define R2b_ 0
#define R2c_ 0
#define ra 7
#define rb 0
#define rc 2*/
#endif
#else
/* Rugby *****/
#define tsum 80.
#define mnt 80
#define dmax 60
#define R1a 0.05
#define R1a_ 0.05
#define pA 0.8
#define R1c 0.0125
#define R1c_ 0.0125
#define R2a 0.04
#define R2a_ 0.04
#define pB 0.7
#define R2c 0.01
#define R2c_ 0.01
#define ra 5
#define rb 2
#define rc 3
#endif

#define dt (0.1)
#define READ 0
#define WRITE 1

```

```

#define over_dmax 1

double xg,yg,zg;
int p[(dmax+1)*2][(int)(mnt/dt)+1];
double P[(dmax+1)*2][(int)(mnt/dt)+1],P_[(dmax+1)*2][(int)(mnt/dt)+1];

FILE *fp;

int paint(int,double,int,int);
double Pf(int,double,double,int),Pf_(int,double,double,int);
void figs(int);

int main(int argc,unsigned char **argv)
{
int fnum;

fp=fopen("bbb.bin","wb");
fprintf(fp," 1705_m.3:\n");

if(LW==0){
fnum=1;
fprintf(fp," mnt=%f\n",(double)mnt);

figs(fnum);
}
else{
fnum=2;
fprintf(fp," te=%f\n",(double)(tsum-mnt));

figs(fnum);
}

fclose(fp);

return 0;
}/** main **/

int paint(int i,double t,int val,int RW)
{
int j;

j=t/dt;

if(RW==0) return p[dmax+i][j];

```

```

else                p[dmax+i][j]=val;

return val;
}/** paint **/

double Pf_(int i,double t,double val,int RW)
{
int j;

j=t/dt;

if(RW==0) return P_[dmax+i][j];
else        P_[dmax+i][j]=val;

return val;
}/** Pf_ **/

double Pf(int i,double t,double val,int RW)
{
int j;

j=t/dt;

if(RW==0) return P[dmax+i][j];
else        P[dmax+i][j]=val;

return val;
}/** Pf **/

void figs(int fnum)
{
int i,d,D,Dp1,Dm1;
int Dp7,Dp5,Dp3,Dp2,Dm7,Dm5,Dm3,Dm2;
double t,/*dt,*/r,r_,val[7];

if(abs(fnum)==1){
/*fp=fopen("bbb.bin","wb");*/

/*999*/
Pf(0,0,1,WRITE);
for(d=1;d<=dmax;d++){
Pf(d,0,0,WRITE);
Pf(-d,0,0,WRITE);

```



```

}
#ifdef Rugby
/* Soccer */
#ifdef Multi
r=1-(R1a*dt+R2a*dt);
#else
r=1-(R1a*dt+R1b*dt+R1c*dt+R2a*dt+R2b*dt+R2c*dt);
#endif
#else
/* Rugby *****/
r=1-(R1a*dt*pA+R1a*dt*(1-pA)+/*rr3A*/R1c*dt+R2a*dt*pB+R2a*dt*(1-pB)
+/*rr3B*/R2c*dt);
#endif

i=1;
for(t=dt;;t+=dt){

D=0;
#ifdef Rugby
/* Soccer */
#ifdef Multi
val[0]=Pf(D-1,t-dt,-1,READ)*R1a*dt+Pf(D,t-dt,-1,READ)*r+Pf(D+1,t-dt,-1,READ)*R2a*dt;
#else
val[0]=Pf(D-ra,t-dt,-1,READ)*R1a*dt+Pf(D-rb,t-dt,-1,READ)*R1b*dt
+Pf(D-rc,t-dt,-1,READ)*R1c*dt
+Pf(D,t-dt,-1,READ)*r
+Pf(D+ra,t-dt,-1,READ)*R2a*dt+Pf(D+rb,t-dt,-1,READ)*R2b*dt
+Pf(D+rc,t-dt,-1,READ)*R2c*dt;
#endif
#else
/* Rugby *****/
val[0]=Pf(D-(ra+rb),t-dt,-1,READ)*R1a*dt*pA+Pf(D-ra,t-dt,-1,READ)*R1a*dt*(1-pA)
+Pf(D-rc,t-dt,-1,READ)*/*rr3A*/R1c*dt
+Pf(D,t-dt,-1,READ)*r
+Pf(D+(ra+rb),t-dt,-1,READ)*R2a*dt*pB+Pf(D+ra,t-dt,-1,READ)*R2a*dt*(1-pB)
+Pf(D+rc,t-dt,-1,READ)*/*rr3B*/R2c*dt;
#endif
Pf(D,t,val[0],WRITE);
xg=0;
xg+=Pf(D,t,-1,READ);

for(d=1;d<=dmax;d++){

D=d;
#ifdef Rugby
/* Soccer */

```

```

#ifndef Multi
Dp1=D+1;if(Dp1>dmax) Dp1=dmax;
val[0]=Pf(D-1,t-dt,-1,READ)*R1a*dt+Pf(D,t-dt,-1,READ)*r+Pf(D+1,t-dt,-1,READ)*R2a*dt;
#else
Dp3=D+ra;Dp2=D+rb;Dp1=D+rc;
/*if(Dp3>dmax){
Dp3=dmax;
Dp2=Dp3-(ra-rb);
Dp1=Dp3-(ra-rc);
}*/
if(Dp3>dmax) Dp3=dmax;
if(Dp2>dmax) Dp2=dmax;
if(Dp1>dmax) Dp1=dmax;
val[0]=Pf(D-ra,t-dt,-1,READ)*R1a*dt+Pf(D-rb,t-dt,-1,READ)*R1b*dt
+Pf(D-rc,t-dt,-1,READ)*R1c*dt
+Pf(D,t-dt,-1,READ)*r
+Pf(/*D+3*/Dp3,t-dt,-1,READ)*R2a*dt+Pf(/*D+2*/Dp2,t-dt,-1,READ)*R2b*dt
+Pf(/*D+1*/Dp1,t-dt,-1,READ)*R2c*dt;
#endif
#else
/* Rugby *****/
Dp7=D+(ra+rb);Dp5=D+ra;Dp3=D+rc;
/*if(Dp7>dmax){
Dp7=dmax;
Dp5=Dp7-rb;
Dp3=Dp7-(ra+rb-rc);
}*/
if(Dp7>dmax) Dp7=dmax;
if(Dp5>dmax) Dp5=dmax;
if(Dp3>dmax) Dp3=dmax;
val[0]=Pf(D-(ra+rb),t-dt,-1,READ)*R1a*dt*pA+Pf(D-ra,t-dt,-1,READ)*R1a*dt*(1-pA)
+Pf(D-rc,t-dt,-1,READ)*/*rr3A*/R1c*dt
+Pf(D,t-dt,-1,READ)*r
+Pf(/*D+7*/Dp7,t-dt,-1,READ)*R2a*dt*pB+Pf(/*D+5*/Dp5,t-dt,-1,READ)*R2a*dt*(1-pB)
+Pf(/*D+3*/Dp3,t-dt,-1,READ)*/*rr3B*/R2c*dt;
#endif
Pf(D,t,val[0],WRITE);
xg+=Pf(D,t,-1,READ);

D=-d;
#ifndef Rugby
/* Soccer */
#ifndef Multi
Dm1=D-1;if(Dm1<-dmax) Dm1=-dmax;
val[0]=Pf(/*D-1*/Dm1,t-dt,-1,READ)*R1a*dt+Pf(D,t-dt,-1,READ)*r

```

```

+Pf(D+1,t-dt,-1,READ)*R2a*dt;
#else
Dm3=D-ra;Dm2=D-rb;Dm1=D-rc;
/*if(Dm3<-dmax){
Dm3=-dmax;
Dm2=Dm3+(ra-rb);
Dm1=Dm3+(ra-rc);
}*/
if(Dm3<-dmax) Dm3=-dmax;
if(Dm2<-dmax) Dm2=-dmax;
if(Dm1<-dmax) Dm1=-dmax;
val[0]=Pf(/*D-3*/Dm3,t-dt,-1,READ)*R1a*dt+Pf(/*D-2*/Dm2,t-dt,-1,READ)*R1b*dt
+Pf(/*D-1*/Dm1,t-dt,-1,READ)*R1c*dt
+Pf(D,t-dt,-1,READ)*r
+Pf(D+ra,t-dt,-1,READ)*R2a*dt+Pf(D+rb,t-dt,-1,READ)*R2b*dt
+Pf(D+rc,t-dt,-1,READ)*R2c*dt;
#endif
#else
/* Rugby *****/
Dm7=D-(ra+rb);Dm5=D-ra;Dm3=D-rc;
/*if(Dm7<-dmax){
Dm7=-dmax;
Dm5=Dm7+rb;
Dm3=Dm7+(ra+rb-rc);
}*/
if(Dm7<-dmax) Dm7=-dmax;
if(Dm5<-dmax) Dm5=-dmax;
if(Dm3<-dmax) Dm3=-dmax;
val[0]=Pf(/*D-7*/Dm7,t-dt,-1,READ)*R1a*dt*pA+Pf(/*D-5*/Dm5,t-dt,-1,READ)*R1a*dt*(1-pA)
+Pf(/*D-3*/Dm3,t-dt,-1,READ)*/*rr3A*/R1c*dt
+Pf(D,t-dt,-1,READ)*r
+Pf(D+(ra+rb),t-dt,-1,READ)*R2a*dt*pB+Pf(D+ra,t-dt,-1,READ)*R2a*dt*(1-pB)
+Pf(D+rc,t-dt,-1,READ)*/*rr3B*/R2c*dt;
#endif
Pf(D,t,val[0],WRITE);
xg+=Pf(D,t,-1,READ);
}/**for(d)**/

```

```

fprintf(fp," tr=%4.1f\n",t);
for(d=-dmax;d<=dmax;d++){
/*if(d<0 && paint(d,t,-1,READ)==1)
fprintf(fp," d=%3dm%.3f",d,Pf(d,t,-1,READ));
else*/
fprintf(fp," d=%3d:%.3f",d,Pf(d,t,-1,READ));

```

```

if(d==-4 || d==3) fprintf(fp,"\n");
}
/*fprintf(fp,"\n sum:%f",xg);*/
fprintf(fp,"\n");

if(i==(int)(mnt/dt)) break;
i++;
}/**for(t)**/

/*fclose(fp);*/
}/**else if(fnum)*****/
else if(abs(fnum)==2){
/*fp=fopen("bbb.bin","wb");*/

/*999*/
Pf(0,mnt/*tsum*/,0.5,WRITE);
Pf_(0,mnt/*tsum*/,0.5,WRITE);
for(d=1;d<=dmax;d++){
Pf(d,mnt/*tsum*/,1,WRITE);
Pf_(d,mnt/*tsum*/,1,WRITE);
Pf(-d,mnt/*tsum*/,0,WRITE);
Pf_(-d,mnt/*tsum*/,0,WRITE);
}
#ifdef Rugby
/* Soccer */
#ifdef Multi
r=1-(R1a*dt+R2a*dt);
r_=1-(R1a_*dt+R2a_*dt);
#else
r=1-(R1a*dt+R1b*dt+R1c*dt+R2a*dt+R2b*dt+R2c*dt);
r_=1-(R1a_*dt+R1b_*dt+R1c_*dt+R2a_*dt+R2b_*dt+R2c_*dt);
#endif
#else
/* Rugby *****/
r=1-(R1a*dt*pA+R1a*dt*(1-pA)+/*rr3A*/R1c*dt+R2a*dt*pB+R2a*dt*(1-pB)+/*rr3B*/R2c*dt);
r_=1-(R1a_*dt*pA+R1a_*dt*(1-pA)+/*rr3A_*/R1c_*dt+R2a_*dt*pB+R2a_*dt*(1-pB)
+/*rr3B_*/R2c_*dt);
#endif

i=mnt/dt-1;                /* -1 <=> -dt */
for(t=mnt/*tsum*/-dt;;t-=dt){

D=0;
#ifdef Rugby

```

```

/* Soccer */
#ifdef Multi
val [0]=Pf(D+1,t+dt,-1,READ)*R1a*dt+Pf(D,t+dt,-1,READ)*r+Pf(D-1,t+dt,-1,READ)*R2a*dt;
if(TCTCS==1)
val [1]=Pf_(D+1,t+dt,-1,READ)*R1a_*dt+Pf_(D,t+dt,-1,READ)*r_
+Pf_(D-1,t+dt,-1,READ)*R2a_*dt;
else if(TCTCS==2)
val [1]=Pf(D+1,t+dt,-1,READ)*R1a_*dt+Pf(D,t+dt,-1,READ)*r_
+Pf(D-1,t+dt,-1,READ)*R2a_*dt;
#else
val [0]=Pf(D+ra,t+dt,-1,READ)*R1a*dt+Pf(D+rb,t+dt,-1,READ)*R1b*dt
+Pf(D+rc,t+dt,-1,READ)*R1c*dt
+Pf(D,t+dt,-1,READ)*r
+Pf(D-ra,t+dt,-1,READ)*R2a*dt+Pf(D-rb,t+dt,-1,READ)*R2b*dt
+Pf(D-rc,t+dt,-1,READ)*R2c*dt;
if(TCTCS==1)
val [1]=Pf_(D+ra,t+dt,-1,READ)*R1a_*dt+Pf_(D+rb,t+dt,-1,READ)*R1b_*dt
+Pf_(D+rc,t+dt,-1,READ)*R1c_*dt
+Pf_(D,t+dt,-1,READ)*r_
+Pf_(D-ra,t+dt,-1,READ)*R2a_*dt+Pf_(D-rb,t+dt,-1,READ)*R2b_*dt
+Pf_(D-rc,t+dt,-1,READ)*R2c_*dt;
else if(TCTCS==2)
val [1]=Pf(D+ra,t+dt,-1,READ)*R1a_*dt+Pf(D+rb,t+dt,-1,READ)*R1b_*dt
+Pf(D+rc,t+dt,-1,READ)*R1c_*dt
+Pf(D,t+dt,-1,READ)*r_
+Pf(D-ra,t+dt,-1,READ)*R2a_*dt+Pf(D-rb,t+dt,-1,READ)*R2b_*dt
+Pf(D-rc,t+dt,-1,READ)*R2c_*dt;
#endif
#else
/* Rugby *****/
val [0]=Pf(D+(ra+rb),t+dt,-1,READ)*R1a*dt*pA+Pf(D+ra,t+dt,-1,READ)*R1a*dt*(1-pA)
+Pf(D+rc,t+dt,-1,READ)*/*rr3A*/R1c*dt
+Pf(D,t+dt,-1,READ)*r
+Pf(D-(ra+rb),t+dt,-1,READ)*R2a*dt*pB+Pf(D-ra,t+dt,-1,READ)*R2a*dt*(1-pB)
+Pf(D-rc,t+dt,-1,READ)*/*rr3B*/R2c*dt;
if(TCTCS==1)
val [1]=Pf_(D+(ra+rb),t+dt,-1,READ)*R1a_*dt*pA+Pf_(D+ra,t+dt,-1,READ)*R1a_*dt*(1-pA)
+Pf_(D+rc,t+dt,-1,READ)*/*rr3A_*/R1c_*dt
+Pf_(D,t+dt,-1,READ)*r_
+Pf_(D-(ra+rb),t+dt,-1,READ)*R2a_*dt*pB+Pf_(D-ra,t+dt,-1,READ)*R2a_*dt*(1-pB)
+Pf_(D-rc,t+dt,-1,READ)*/*rr3B_*/R2c_*dt;
else if(TCTCS==2)
val [1]=Pf(D+(ra+rb),t+dt,-1,READ)*R1a_*dt*pA+Pf(D+ra,t+dt,-1,READ)*R1a_*dt*(1-pA)
+Pf(D+rc,t+dt,-1,READ)*/*rr3A_*/R1c_*dt
+Pf(D,t+dt,-1,READ)*r_
+Pf(D-(ra+rb),t+dt,-1,READ)*R2a_*dt*pB+Pf(D-ra,t+dt,-1,READ)*R2a_*dt*(1-pB)

```

```

+Pf(D-rc,t+dt,-1,READ)*/*rr3B_*/R2c_*dt;
#endif
Pf(D,t,val[0],WRITE);
Pf_(D,t,val[1],WRITE);

for(d=1;d<=dmax;d++){

D=d;
#ifdef Rugby
/* Soccer */
#ifdef Multi
Dp1=D+1;if(Dp1>dmax) Dp1=dmax;
val[0]=Pf(/*D+1*/Dp1,t+dt,-1,READ)*R1a*dt+Pf(D,t+dt,-1,READ)*r
+Pf(D-1,t+dt,-1,READ)*R2a*dt;
if(TCTCS==1)
val[1]=Pf_(/*D+1*/Dp1,t+dt,-1,READ)*R1a_*dt+Pf_(D,t+dt,-1,READ)*r_
+Pf_(D-1,t+dt,-1,READ)*R2a_*dt;
else if(TCTCS==2)
val[1]=Pf(/*D+1*/Dp1,t+dt,-1,READ)*R1a_*dt+Pf(D,t+dt,-1,READ)*r_
+Pf(D-1,t+dt,-1,READ)*R2a_*dt;
#else
Dp3=D+ra;Dp2=D+rb;Dp1=D+rc;

#if over_dmax==0
if(Dp3>dmax){
Dp3=dmax;
Dp2=Dp3-(ra-rb);
Dp1=Dp3-(ra-rc);
}
#else
if(Dp3>dmax) Dp3=dmax;
if(Dp2>dmax) Dp2=dmax;
if(Dp1>dmax) Dp1=dmax;
#endif

val[0]=Pf(/*D+3*/Dp3,t+dt,-1,READ)*R1a*dt+Pf(/*D+2*/Dp2,t+dt,-1,READ)*R1b*dt
+Pf(/*D+1*/Dp1,t+dt,-1,READ)*R1c*dt
+Pf(D,t+dt,-1,READ)*r
+Pf(D-ra,t+dt,-1,READ)*R2a*dt+Pf(D-rb,t+dt,-1,READ)*R2b*dt
+Pf(D-rc,t+dt,-1,READ)*R2c*dt;
if(TCTCS==1)
val[1]=Pf_(/*D+3*/Dp3,t+dt,-1,READ)*R1a_*dt+Pf_(/*D+2*/Dp2,t+dt,-1,READ)*R1b_*dt
+Pf_(/*D+1*/Dp1,t+dt,-1,READ)*R1c_*dt
+Pf_(D,t+dt,-1,READ)*r_
+Pf_(D-ra,t+dt,-1,READ)*R2a_*dt+Pf_(D-rb,t+dt,-1,READ)*R2b_*dt
+Pf_(D-rc,t+dt,-1,READ)*R2c_*dt;

```

```

else if(TCTCS==2)
val[1]=Pf(/*D+3*/Dp3,t+dt,-1,READ)*R1a_*dt+Pf(/*D+2*/Dp2,t+dt,-1,READ)*R1b_*dt
+Pf(/*D+1*/Dp1,t+dt,-1,READ)*R1c_*dt
+Pf(D,t+dt,-1,READ)*r_
+Pf(D-ra,t+dt,-1,READ)*R2a_*dt+Pf(D-rb,t+dt,-1,READ)*R2b_*dt
+Pf(D-rc,t+dt,-1,READ)*R2c_*dt;
#endif
#else
/* Rugby *****/
Dp7=D+(ra+rb);Dp5=D+ra;Dp3=D+rc;

#if over_dmax==0
if(Dp7>dmax){
Dp7=dmax;
Dp5=Dp7-rb;
Dp3=Dp7-(ra+rb-rc);
}
#else
if(Dp7>dmax) Dp7=dmax;
if(Dp5>dmax) Dp5=dmax;
if(Dp3>dmax) Dp3=dmax;
#endif

val[0]=Pf(/*D+7*/Dp7,t+dt,-1,READ)*R1a_*dt*pA+Pf(/*D+5*/Dp5,t+dt,-1,READ)*R1a_*dt*(1-pA)
+Pf(/*D+3*/Dp3,t+dt,-1,READ)*/*rr3A*/R1c_*dt
+Pf(D,t+dt,-1,READ)*r
+Pf(D-(ra+rb),t+dt,-1,READ)*R2a_*dt*pB+Pf(D-ra,t+dt,-1,READ)*R2a_*dt*(1-pB)
+Pf(D-rc,t+dt,-1,READ)*/*rr3B*/R2c_*dt;
if(TCTCS==1)
val[1]=Pf_(/*D+7*/Dp7,t+dt,-1,READ)*R1a_*dt*pA
+Pf_(/*D+5*/Dp5,t+dt,-1,READ)*R1a_*dt*(1-pA)
+Pf_(/*D+3*/Dp3,t+dt,-1,READ)*/*rr3A_*/R1c_*dt
+Pf_(D,t+dt,-1,READ)*r
+Pf_(D-(ra+rb),t+dt,-1,READ)*R2a_*dt*pB+Pf_(D-ra,t+dt,-1,READ)*R2a_*dt*(1-pB)
+Pf_(D-rc,t+dt,-1,READ)*/*rr3B_*/R2c_*dt;
else if(TCTCS==2)
val[1]=Pf(/*D+7*/Dp7,t+dt,-1,READ)*R1a_*dt*pA
+Pf(/*D+5*/Dp5,t+dt,-1,READ)*R1a_*dt*(1-pA)
+Pf(/*D+3*/Dp3,t+dt,-1,READ)*/*rr3A_*/R1c_*dt
+Pf(D,t+dt,-1,READ)*r
+Pf(D-(ra+rb),t+dt,-1,READ)*R2a_*dt*pB+Pf(D-ra,t+dt,-1,READ)*R2a_*dt*(1-pB)
+Pf(D-rc,t+dt,-1,READ)*/*rr3B_*/R2c_*dt;
#endif
Pf(D,t,val[0],WRITE);
Pf_(D,t,val[1],WRITE);

```

```

D=-d;
#ifdef Rugby
/* Soccer */
#ifdef Multi
Dm1=D-1;if(Dm1<-dmax) Dm1=-dmax;
val[0]=Pf(D+1,t+dt,-1,READ)*R1a*dt+Pf(D,t+dt,-1,READ)*r
+Pf(/*D-1*/Dm1,t+dt,-1,READ)*R2a*dt;
if(TCTCS==1)
val[1]=Pf_(D+1,t+dt,-1,READ)*R1a_*dt+Pf_(D,t+dt,-1,READ)*r_
+Pf_(/*D-1*/Dm1,t+dt,-1,READ)*R2a_*dt;
else if(TCTCS==2)
val[1]=Pf(D+1,t+dt,-1,READ)*R1a_*dt+Pf(D,t+dt,-1,READ)*r_
+Pf(/*D-1*/Dm1,t+dt,-1,READ)*R2a_*dt;
#else
Dm3=D-ra;Dm2=D-rb;Dm1=D-rc;

#if over_dmax==0
if(Dm3<-dmax){
Dm3=-dmax;
Dm2=Dm3+(ra-rb);
Dm1=Dm3+(ra-rc);
}
#else
if(Dm3<-dmax) Dm3=-dmax;
if(Dm2<-dmax) Dm2=-dmax;
if(Dm1<-dmax) Dm1=-dmax;
#endif

val[0]=Pf(D+ra,t+dt,-1,READ)*R1a*dt+Pf(D+rb,t+dt,-1,READ)*R1b*dt
+Pf(D+rc,t+dt,-1,READ)*R1c*dt
+Pf(D,t+dt,-1,READ)*r
+Pf(/*D-3*/Dm3,t+dt,-1,READ)*R2a*dt+Pf(/*D-2*/Dm2,t+dt,-1,READ)*R2b*dt
+Pf(/*D-1*/Dm1,t+dt,-1,READ)*R2c*dt;
if(TCTCS==1)
val[1]=Pf_(D+ra,t+dt,-1,READ)*R1a_*dt+Pf_(D+rb,t+dt,-1,READ)*R1b_*dt
+Pf_(D+rc,t+dt,-1,READ)*R1c_*dt
+Pf_(D,t+dt,-1,READ)*r_
+Pf_(/*D-3*/Dm3,t+dt,-1,READ)*R2a_*dt+Pf_(/*D-2*/Dm2,t+dt,-1,READ)*R2b_*dt
+Pf_(/*D-1*/Dm1,t+dt,-1,READ)*R2c_*dt;
else if(TCTCS==2)
val[1]=Pf(D+ra,t+dt,-1,READ)*R1a_*dt+Pf(D+rb,t+dt,-1,READ)*R1b_*dt
+Pf(D+rc,t+dt,-1,READ)*R1c_*dt
+Pf(D,t+dt,-1,READ)*r_
+Pf(/*D-3*/Dm3,t+dt,-1,READ)*R2a_*dt+Pf(/*D-2*/Dm2,t+dt,-1,READ)*R2b_*dt
+Pf(/*D-1*/Dm1,t+dt,-1,READ)*R2c_*dt;

```



```

#endif
#else
/* Rugby *****/
Dm7=D-(ra+rb);Dm5=D-ra;Dm3=D-rc;

#if over_dmax==0
if(Dm7<-dmax){
Dm7=-dmax;
Dm5=Dm7+rb;
Dm3=Dm7+(ra+rb-rc);
}
#else
if(Dm7<-dmax) Dm7=-dmax;
if(Dm5<-dmax) Dm5=-dmax;
if(Dm3<-dmax) Dm3=-dmax;
#endif

val[0]=Pf(D+(ra+rb),t+dt,-1,READ)*R1a*dt*pA+Pf(D+ra,t+dt,-1,READ)*R1a*dt*(1-pA)
+Pf(D+rc,t+dt,-1,READ)*/*rr3A*/R1c*dt
+Pf(D,t+dt,-1,READ)*r
+Pf(/*D-7*/Dm7,t+dt,-1,READ)*R2a*dt*pB+Pf(/*D-5*/Dm5,t+dt,-1,READ)*R2a*dt*(1-pB)
+Pf(/*D-3*/Dm3,t+dt,-1,READ)*/*rr3B*/R2c*dt;
if(TCTCS==1)
val[1]=Pf_(D+(ra+rb),t+dt,-1,READ)*R1a_*dt*pA+Pf_(D+ra,t+dt,-1,READ)*R1a_*dt*(1-pA)
+Pf_(D+rc,t+dt,-1,READ)*/*rr3A_*/R1c_*dt
+Pf_(D,t+dt,-1,READ)*r_
+Pf_(/*D-7*/Dm7,t+dt,-1,READ)*R2a_*dt*pB+Pf_(/*D-5*/Dm5,t+dt,-1,READ)*R2a_*dt*(1-pB)
+Pf_(/*D-3*/Dm3,t+dt,-1,READ)*/*rr3B_*/R2c_*dt;
else if(TCTCS==2)
val[1]=Pf(D+(ra+rb),t+dt,-1,READ)*R1a_*dt*pA+Pf(D+ra,t+dt,-1,READ)*R1a_*dt*(1-pA)
+Pf(D+rc,t+dt,-1,READ)*/*rr3A_*/R1c_*dt
+Pf(D,t+dt,-1,READ)*r_
+Pf(/*D-7*/Dm7,t+dt,-1,READ)*R2a_*dt*pB+Pf(/*D-5*/Dm5,t+dt,-1,READ)*R2a_*dt*(1-pB)
+Pf(/*D-3*/Dm3,t+dt,-1,READ)*/*rr3B_*/R2c_*dt;
#endif

/*99*/
if(TCTCS>0){
if(val[0]<val[1]) {val[0]=val[1];paint(D,t,1,WRITE);} else {paint(D,t,0,WRITE);}
}

Pf(D,t,val[0],WRITE);
Pf_(D,t,val[1],WRITE);
}/**for(d)**/

```

```
fprintf(fp," te=%4.1f\n",t);
for(d=-dmax;d<=dmax;d++){
if(paint(d,t,-1,READ)==1)
fprintf(fp," d=%3dm%.3f",d,Pf_(d,t,-1,READ));
else
fprintf(fp," d=%3d:%.3f",d,Pf(d,t,-1,READ));
if(d==-4 || d==3) fprintf(fp,"\n");
}
fprintf(fp,"\n");
```

```
if(i==0) break;
i--;
}/**for(t)**/
```

```
/*fclose(fp);*/
}/**else if(fnum)**/
else if(abs(fnum)==3){
```

```
}/**else if(fnum)**/
}/** figs **/
```