Mach Learn (2019) 108:127–147
https://doi.org/10.1007/s10994-018-5725-1

Probabilistic movement models and zones of control

Ulf Brefeld[1] · Jan Lasek[2] · Sebastian Mair[1]

Received: 21 June 2017 / Accepted: 3 June 2018 / Published online: 29 June 2018

Abstract Coordinated movements of players are key to success in team sports. However, traditional models for player movements are based on unrealistic assumptions and their analysis is prone to errors. As a remedy, we propose to estimate individual movement models from positional data and show how to turn these estimates into accurate and realistic zones of control. Our approach accounts for characteristic traits of players, scales with large amounts of data, and can be efficiently computed in a distributed fashion. We report on empirical results.

Keywords Positional data · Movement models · Zones of control · Soccer

1 Introduction

Player coordination is perhaps the most important aspect in team sports. In soccer, for example, collective movements are inalienable for controlling the midfield, counter attacks, or effective pressing (Taki and Hasegawa 2000; Fonseca et al. 2012; Gudmundsson and Wolle 2014; Horton et al. 2015). Therefore, models that quantify the probability that a player attains a certain position in a given time are crucial. Such models are called *movement models*.

Editors: Philippe Lopes, Werner Dubitzky, Daniel Berrar, and Jesse Davis.

✉ Jan Lasek
janek.lasek@gmail.com

Ulf Brefeld
brefeld@leuphana.de

Sebastian Mair
mair@leuphana.de

[1] Leuphana University, Lüneburg, Germany

[2] Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland

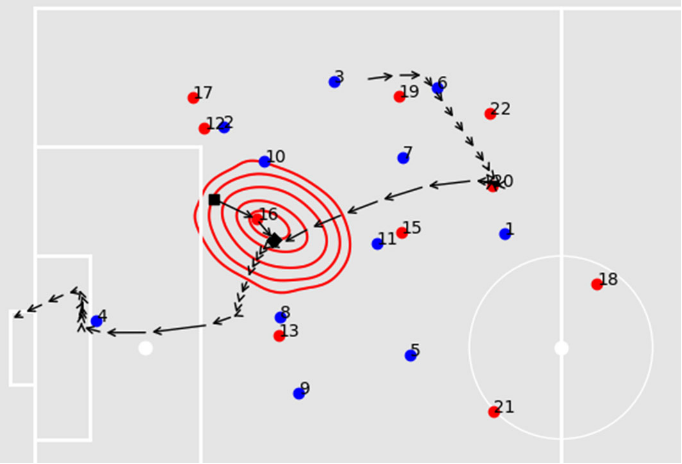

Fig. 1 An exemplary movement model shown as contour plot in red around player 16. The black squares indicate the position of player 16 a second in the past and in the future. The black arrows are showing 10 s of the ball movement

Traditional movement models ground on the assumption that players are able to move in all directions equally fast and ignore velocities (Taki et al. 1996; Taki and Hasegawa 2000; Fonseca et al. 2012), leading to implausible Voronoi-like tessellations (Voronoi 1908) of the pitch. More sophisticated models incorporate some basic laws of physics but either suffer from unrealistic assumptions or remain intellectual pastimes (Taki and Hasegawa 2000; Fujimura and Sugihara 2005; Gudmundsson and Wolle 2014). All existing approaches treat every player the same by assuming that a single movement model serves all players equally well, hence, ignoring individual differences between players.

Consequences arise for applications that build upon player movement such as the computation of *zones of control* (sometimes also called dominant regions). The zone that is controlled by a player is characterized by her being the person on the pitch to attain any position within this region first (Taki and Hasegawa 2000). The underlying idea is that if the ball falls inside a player's zone of control, she will likely be able to bring the ball under control after receiving it and the more space a team controls, the more dominant they are.

In this paper, we propose to estimate individual movement models from positional data. Our probabilistic approach leverages positions, directions, and velocities of a player at observed timestamps and returns a distribution of all reachable positions in a given time. Figure 1 shows an example. We present an efficient computational schema for processing positional data at large scales and show how to turn the probabilistic movement models into zones of control. Compared to traditional one-serves-all methods, our approach leads to realistic movement models, which in turn lead to realistic zones of control.

The remainder is organized as follows. Section 2 reviews related work. Section 3 presents the estimation of individual movement models and Sect. 4 the computation of the resulting zones of control. Section 5 provides a discussion and Sect. 6 concludes.

2 Related work

Trajectory analyses are often carried out for wearable devices like smart phones, accelerometers, or gyroscopes (Zheng 2015; Mazimpaka and Timpf 2016). Often, the trajectories serve only as proxies for a higher level research question such as the identification of road defects (see, e.g., Byrne et al. 2013; Mohan et al. 2008), discrimination of drivers by insurance companies (Paefgen et al. 2011), or activity recognition (Avci et al. 2010; Lasek and Gagolewski 2015).

Similarly, trajectory data in sports is used to identify movement patterns. At an individual level, Zhao et al. (2016) use Gaussian processes to model velocity (flow) of athletes in ski races. Laube et al. (2005) propose to analyze relative motions and different temporal patterns across many subjects. As an exemplary application, the authors analyze positional data to retrieve patterns from coordinated team motions. The problem of pattern identification in groups of moving objects is also studied by Gottfried (2008, 2011). The author proposes qualitative descriptions of motion patterns using a set of atomic motions as building blocks to analyze and describe more complex behaviors; Sprado and Gottfried (2009) apply this idea to robocup and soccer games. Knauf et al. (2016) propose spatio-temporal convolution kernels as a similarity measure over time and space and identify game initiations and offensive patterns using a clustering approach. Similarly, Janetzko et al. (2014) group attacking patterns of strikers. Generally, frequent patterns in multi-trajectory data can also be found using episode mining algorithms (Haase and Brefeld 2014).

Zhang et al. (2016) visualize time interval data to analyze player and team performance. They include a variety of features ranging from player velocities and ball possession as the team dominance metrics. Other methods include, for example, estimating the probabilities of a shot being made (Link et al. 2016; Harmon et al. 2016). Generally, the application of neural networks to player trajectories, represented either as sequences or images, render the need for engineering hand-crafted features unnecessary and may thus be beneficial in situations where sufficient statistics are unknown or difficult to obtain, as for analyzing player positioning. For instance, Zheng et al. (2016) and Le et al. (2017) propose to model player trajectories with recurrent neural networks for player positioning in basketball and soccer. Similarly, convolutional neural networks are used by Harmon et al. (2016) to estimate the probability of scoring opportunities. Memmert et al. (2016) and Gudmundsson and Horton (2017) provide a general overview of positional data applications in team sports. Other interesting applications include pass quality evaluation (Brooks et al. 2016) or injury prediction (Rossi et al. 2017).

Taki and Hasegawa (2000) propose a movement model that is based on a player's current speed, her direction, and an acceleration profile along different directions. The authors discuss the dependency of acceleration on velocity and direction and also emphasize that the acceleration decreases with increasing speed. Unfortunately, the authors ignore physical details and focus on a very basic and unrealistic version of their model, in which a player is able to move in all directions with the same acceleration; hence, accepting the consequence of unbounded velocities. Fujimura and Sugihara (2005) extend this approach by adding a resistive force to prevent velocities to grow infinitely. Thus, the two approaches drastically simplify physical laws to model player movements. Note that both also constitute one-serves-all approaches as the model is not personalized to account for individual differences between players. Recently, Gudmundsson and Wolle (2014) sketch how such an individual movement model could be estimated from data. They suggest approximating a player's *reachable region* at time t by constructing a convex polygon for all historic points she reached within this time given her actual position. However, they leave it a play of thoughts and do not present technical or algorithmic details of their approach.

Once a movement model is established, it serves as a foundation for various applications in the analysis of matches. Perhaps the most important one being the computation of *zones of control*, or, alternatively, *dominant regions*. This concept has been introduced by Taki and

Hasegawa (2000) as the part of the pitch that can be attained by a player before all others. Consequently, zones of control are necessary to compute and evaluate pass quality and success (Taki and Hasegawa 2000; Nakanishi et al. 2009; Gudmundsson and Wolle 2014; Horton et al. 2015), pressing (Taki and Hasegawa 2000), as well as the analysis of team behavior and interaction (Fonseca et al. 2012), or organization and positioning in both offense and defense (Ueda et al. 2014).

3 Estimating individual movement models

3.1 Preliminaries

Let $\mathbf{p}^k = (x_t^k, y_t^k)_{t \in \mathbb{R}_{\geq 0}}$ be the trajectory of player k describing her position in area $F \subset \mathbb{R}^2$ and let $\mathbf{v}_t^k \in \mathbb{R}^2$ be her velocity vector at time $t \in \mathbb{R}_{\geq 0}$ with its magnitude (speed) $v_t^k = \|\mathbf{v}_t^k\|_2$, where $\| \cdot \|_2$ denotes the ℓ_2-norm.[1] The time index t is typically discrete as samples are generated with equidistant timestamps $t_1, t_2, \ldots, t_n$, where $t_{i+1} - t_i = \tau > 0$ is fixed. The trajectories and the associated velocities form the dataset $\mathcal{D} = \{(\mathbf{p}_{t_i}, v_{t_i})\}_{i=1}^n$. The goal is to generate a probabilistic model of the player's whereabouts in time horizon $t_\Delta > 0$ given her current position $\mathbf{p}_t^k$ and velocity $\mathbf{v}_t^k$:

$$\mathbb{P}_{t_\Delta}^k \left((x, y) \mid \mathbf{p}_t^k, \mathbf{v}_t^k \right).$$

To not clutter notation unnecessarily, we discard the player index k whenever possible and focus on data from a single player.

3.2 Existing approaches

Before we introduce the estimation of probabilistic movement models from positional data, we briefly review existing approaches. The simplest model assumes that all players are able to move in all directions equally fast at a constant speed. Thus, there is no acceleration or direction of movement and the resulting zones of control are equal to Voronoi tessellations (Voronoi 1908) of the pitch using the players as center points. This model is referred to as *Voronoi*.

Taki and Hasegawa (2000) improve on this by incorporating the notion of velocity and acceleration. Their model is based on the assumption that every player is able to accelerate in each direction equally fast with the magnitude of $a_{\max} > 0$. Thus, at time $t = 0$ the player begins to move with acceleration $a_{\max}$ in a direction given by the angle $\phi \in [-\pi, \pi)$. Assuming that a player is moving with speed v in the direction of the x-axis, in time t her position $\mathbf{p} = (x, y)$ is given by

$$\mathbf{p} = (x, y) \quad \text{with} \quad \begin{cases} x = \frac{1}{2} a_{\max} \cdot \cos(\phi) \cdot t^2 + vt \\ y = \frac{1}{2} a_{\max} \cdot \sin(\phi) \cdot t^2. \end{cases} \tag{1}$$

In other words, the set of points that can be reached in time t forms a circle centered at $\mathbf{c} \in \mathbb{R}^2$ with radius $r > 0$, where

$$\mathbf{c} = (vt, 0) \quad \text{and} \quad r = \frac{1}{2} a_{\max} t^2,$$

[1] Note that the velocity can be estimated from positional data in case it is not provided directly.

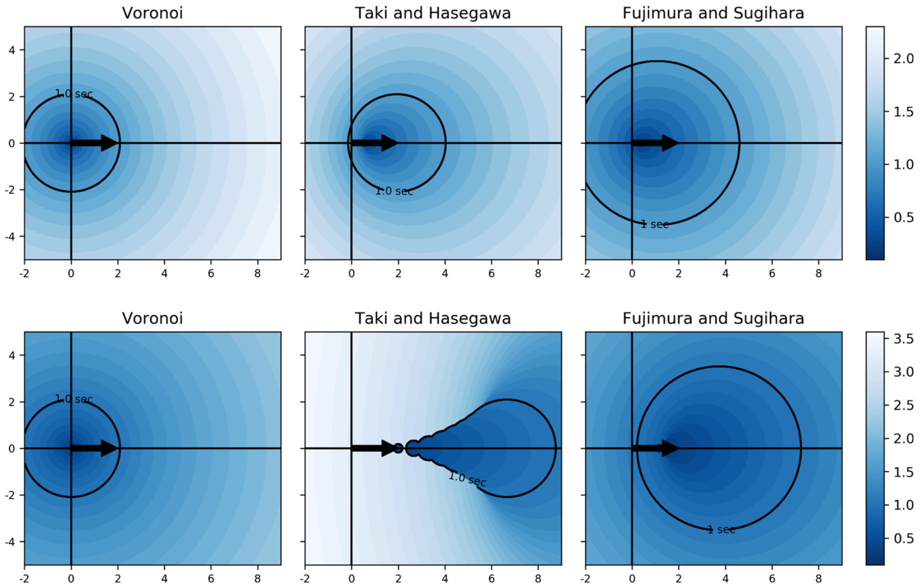

Fig. 2 Movement models for a player running with 7.0 km/h (top row) and 24 km/h (bottom row) in direction of x-axis. Voronoi is computed like Taki & Hasegawa but with a velocity of zero

thus, allowing for unbounded velocities. The model is depicted in Fig. 2. The details of setting the model's parameters are relegated to Sect. 3.7. We refer to this model as *Taki & Hasegawa*.

Fujimura and Sugihara (2005) introduce a resistive force proportional to the current speed to render the movement model more realistic. The resistive force prevents the speed to grow infinitely and even clips it at maximal value $v_{\max} > 0$. Thus, at time $t = 0$, a player accelerates in direction $\phi \in [-\pi, \pi)$ with the underlying assumption that she can exert the maximum speed in any direction. The position $\mathbf{p} = (x, y)$ of the player at time t is given by

$$\mathbf{p} = (x, y) \quad \text{with} \quad \begin{cases} x = v_{\max} \cdot \cos(\phi) \cdot \left(t - \frac{1-\exp(-\alpha t)}{\alpha}\right) + v \cdot \frac{1-\exp(-\alpha t)}{\alpha} \\ y = v_{\max} \cdot \sin(\phi) \cdot \left(t - \frac{1-\exp(-\alpha t)}{\alpha}\right), \end{cases} \tag{2}$$

where v is the initial velocity in the direction of the x-axis and the parameter $\alpha > 0$ is responsible for the resistive force. Hence, the set of points within reach of the player in time t forms a circle centered at $\mathbf{c} \in \mathbb{R}^2$ with radius $r > 0$, where

$$\mathbf{c} = \left(v \cdot \frac{1 - \exp(-\alpha t)}{\alpha}, 0\right) \quad \text{and} \quad r = v_{\max} \cdot \left(t - \frac{1 - \exp(-\alpha t)}{\alpha}\right).$$

The model is referred to as *Fujimura & Sugihara*.

Figure 2 visualizes the existing movement models obtained by Voronoi-based approaches (from left to right). While all models realize similar circular-shaped movements for slowly moving players, differences become significant with increasing velocities. While the Voronoi-based approach yields perfect circles for any velocity, the approach by Fujimura & Sugihara leads to a conical structure assembled by nested circles. Finally, Taki & Hasegawa-based movement models become drop-shaped and oblique conical. Simply by being intrinsically circular for arbitrary velocities, it becomes obvious that the existing models serve only as crude approximations of reality. Intuitively, one would expect an elliptically shaped move-

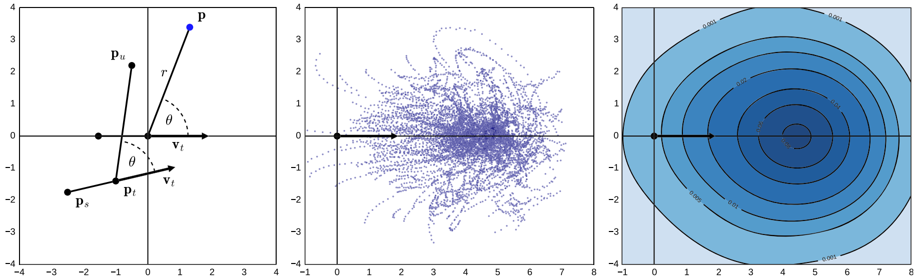

Fig. 3 Left: Illustration of the ψ function. Center and right: Example for a time horizon of $t_\Delta = 1$ s and a player velocity of 14–20 km/h. Position triplets $(\mathbf{p}_s, \mathbf{p}_t, \mathbf{p}_u)$ are used to obtain data samples (center) which are then smoothed using a KDE with Gaussian kernel to obtain the movement model (right)

ment model, and, we will show in the next section that the data-driven models take on elliptical shapes.

3.3 Estimating individual movement models from positional data

We now describe how to compute probabilistic movement models from positional data. The computation is based on triplets $(\mathbf{p}_s, \mathbf{p}_t, \mathbf{p}_u)$ with $s < t < u$, $t - s = t_\delta$ and $u - t = t_\Delta$ that are drawn from a player's trajectory. The coordinates $\mathbf{p}_s$ and $\mathbf{p}_t$ will be used to estimate the direction in which the player moves while $\mathbf{p}_u$ will be used to estimate her ability to move. First, the triplet is transformed by a translation so that point p_t is centered at $(0, 0)$ followed by a rotation so that the vector $\overrightarrow{\mathbf{p}_s\mathbf{p}_t} = (x_t - x_s, y_t - y_s)$ is aligned with the x-axis. This way, the transformed position $\mathbf{p}_u$ describes the point the player reaches assuming her current position is the origin, moving in direction of x-axis with a given speed $v_t = \|\mathbf{v}_t\|_2$. Figure 3 provides an overview of this approach.

Let $(\mathbf{p}_s, \mathbf{p}_t, \mathbf{p}_u)$ be a triplet of positions within a player's trajectory and let ψ be the function that maps such a triplet to (x, y) coordinates,

$$\psi : \mathbb{R}^2 \times \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}^2, \quad (\mathbf{p}_s, \mathbf{p}_t, \mathbf{p}_u) \mapsto \mathbf{p} = (x, y). \tag{3}$$

We obtain the destination point $\mathbf{p} = (x, y)$ using a representation in polar coordinates

$$(x, y) = (r \cdot \cos(\theta),\ r \cdot \sin(\theta)),$$

where θ is a signed angle and r the distance. The angle θ is computed via the following direct calculation

$$\theta = \measuredangle(\overrightarrow{\mathbf{p}_s\mathbf{p}_t}, \overrightarrow{\mathbf{p}_t\mathbf{p}_u}) = \operatorname{atan2}(y_t - y_s, x_t - x_s) - \operatorname{atan2}(y_u - y_t, x_u - x_t) \tag{4}$$

for $\mathbf{p}_s \neq \mathbf{p}_t, \mathbf{p}_t \neq \mathbf{p}_u$, where $\operatorname{atan2}(y, x)$ is a function that yields an angle $\theta \in [-\pi, \pi)$ between point (x, y) and the positive x–axis. The distance is given by

$$r = \|\overrightarrow{\mathbf{p}_t\mathbf{p}_u}\|_2. \tag{5}$$

Figure 3 (left) illustrates how the mapping ψ processes data triplets to derive the position $\mathbf{p}$. Samples of the transformed positions and the associated speed values are collected within the set $\mathcal{S}_{t_\Delta}$. This approach is summarized in Algorithm 1.

Having obtained set $\mathcal{S}_{t_\Delta}$, it is possible to define a probability distribution over possible player whereabouts given her position and initial velocity. This can be done with a two-dimensional kernel density estimate (KDE). Due to practical considerations, we suggest to

Algorithm 1 Computation of movement samples.

```
Input: Data set $\mathcal{D} = \{(\mathbf{p}_{t_i}, v_{t_i})\}_{i=1}^{n}$
Output: Set $\mathcal{S}_{t_\Delta}$ of attained positions in time $t_\Delta$ including initial velocities
for $s < t < u$ s.t. $s = t - t_\delta, u = t + t_\Delta$ do
    $\mathbf{p} = \psi(\mathbf{p}_s, \mathbf{p}_t, \mathbf{p}_u)$        ▷ transformed destination as outlined in Eq. (3)
    $\mathcal{S}_{t_\Delta} = \mathcal{S}_{t_\Delta} \cup \{(\mathbf{p}, v_t)\}$        ▷ append the derived sample to set $\mathcal{S}_{t_\Delta}$
end for
```

discretize the speed range and include it in the model at a lower level of granularity. To this end, we define a subset of points

$$\mathcal{S}_{t_\Delta,V} = \{(\mathbf{p}, v) \mid v \in V\} \subseteq \mathcal{S}_{t_\Delta}$$

for a range of velocity values in the interval $V = [v_{\min}, v_{\max}]$ and compute the individual movement model using a KDE based on samples from this set. We obtain several KDEs depending on different velocity ranges denoted as $\mathbb{P}^{\text{KDE}}_{t_\Delta,V}$. To evaluate the likelihood of attaining a given position $\mathbf{p} \in \mathbb{R}^2$, we use

$$\mathbb{P}_{t_\Delta}(\mathbf{p} \mid \mathbf{p}_t, \mathbf{v}_t) = \mathbb{P}_{t_\Delta}\big(\mathbf{p} \mid \mathbf{p}_t, \mathbf{p}_{t-t_\delta}, v_t\big) = \mathbb{P}^{\text{KDE}}_{t_\Delta,V}\big(\psi(\mathbf{p}_{t-t_\delta}, \mathbf{p}_t, \mathbf{p})\big) \tag{6}$$

for $v_t \in V$. We introduce an extra conditioning on the previous player's position $\mathbf{p}_{t-t_\delta}$ utilized for the estimation of the direction (angle θ) in which the player moves. Figure 3 (center) presents a set of samples collected and Fig. 3 (right) a corresponding movement model based on a KDE.[2]

The model relies on a particular discretization of the speed range $\mathcal{V}$ denoted by $\tilde{\mathcal{V}}$. Analogously, different models are obtained for different values of the time horizon parameter t_Δ. In fact, we are interested in several values of this parameter for different time horizons (of about one second) in a given interval $\mathcal{T}$.

In some cases, the triplets of points used to estimate the model can contain outliers. They may stem from an interruption during a match (e.g., due to a foul or corner kick) or errors in the data collecting process. Hence, triplets containing outliers should be discarded. Finally, given that a player's ability to move should be symmetric with respect to the direction she is facing, the set can be augmented with $(\bar{\mathbf{p}}, v)$ using $\bar{\mathbf{p}} = (x, -y)$ for each sample $(\mathbf{p}, v) \in \mathcal{S}_{t_\Delta,V}$.

3.4 Large-scale movement models

Considering the huge amount of data possibly covering multiple seasons per player, using KDEs to estimate the movement model for a player can be problematic due to efficiency issues. Clearly, the more data points are used for training, the better the KDE and hence the final model. However, more points slow down the prediction and KDE-based movement models may become prohibitive for real-time analysis. An alternative way of defining the movement models is to compute two-dimensional histograms as follows.

We denote the equidistant discretization of an interval $\mathcal{Z}$ by $\tilde{\mathcal{Z}}$, which is of size $n_\mathcal{Z}$. More specifically, $\tilde{\mathcal{Z}}$ is a sequence of $\mathcal{Z}_i = \min(\mathcal{Z}) + i \cdot \Delta$, $i = 0, 1, 2, \ldots, n_\mathcal{Z}$ and $\Delta = \frac{\max(\mathcal{Z}) - \min(\mathcal{Z})}{n_\mathcal{Z}}$. Hence, the ith cell is $[\mathcal{Z}_{i-1}, \mathcal{Z}_i) \subseteq \mathcal{Z}$ and $\mathcal{Z} = [\mathcal{Z}_0, \mathcal{Z}_1) \cup [\mathcal{Z}_1, \mathcal{Z}_2) \cup \cdots \cup [\mathcal{Z}_{n_\mathcal{Z}-1}, \mathcal{Z}_{n_\mathcal{Z}})$. The space $\mathcal{X} \times \mathcal{Y}$ covers the possible whereabouts of a player in a given time horizon. Interval $\mathcal{V}$ contains all possible velocities as introduced in the previous section.

[2] In the figure, a Gaussian-based KDE is used with the bandwidth parameter set to 0.7 [see, e.g., Turlach (1993) for an overview of bandwidth selection methods].

Algorithm 2 Computation of the count matrix A

```
Input: Data set $\mathcal{D} = \{(\mathbf{p}_{t_i}, v_{t_i})\}_{i=1}^{n}$
Output: Matrix $A$ containing all counts.
$A = \mathbf{0}^{n_\mathcal{X} \times n_\mathcal{Y} \times n_\mathcal{V} \times n_\mathcal{T}}$          ▷ initialize empty matrix of counts
for $t_\Delta \in \tilde{\mathcal{T}}$ do
    for $s < t < u$ s.t. $s = t - t_\delta, u = t + t_\Delta$ do
        $(a, b, c, d) = \text{indices}\left(\psi(\mathbf{p}_s, \mathbf{p}_t, \mathbf{p}_u), v_t, t_\Delta\right)$          ▷ outlined in Eq. (7)
        $A_{abcd} = A_{abcd} + 1$          ▷ increment $A_{abcd}$
    end for
end for
```

Finally, $\tilde{\mathcal{T}}$ denotes a sequence of all time horizons of interest within an interval $\mathcal{T}$. Let $\tilde{\mathcal{X}}$, $\tilde{\mathcal{Y}}$, and $\tilde{\mathcal{V}}$ be the discretizations of $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{V}$ with sizes $n_\mathcal{X}$, $n_\mathcal{Y}$, and $n_\mathcal{V}$, respectively. Furthermore, let A be a $n_\mathcal{X} \times n_\mathcal{Y} \times n_\mathcal{V} \times n_\mathcal{T}$ matrix containing the counts of points. Here, entry $A_{abcd} \in \mathbb{N}$ contains the counts for all points within the ath cell in $\mathcal{X}$, the bth cell in $\mathcal{Y}$, the cth speed range in $\mathcal{V}$, and the dth time delta from $\mathcal{T}$. Given a time delta t_Δ, we compute the indices a, b, c, and d for the trajectory point corresponding to the transformed position $\mathbf{p} = (x, y)$ and the speed v as discussed in the previous section. The assignment of the indices is done by the following function:

$$\begin{aligned}\text{indices}: \quad & \mathbb{R}^2 \times \mathbb{R} \times \mathbb{R} \to \mathbb{N}_0^4, \quad \left(\mathbf{p}, v, t_\Delta\right) \mapsto (a, b, c, d), \\ & \text{where} \quad x \in [\mathcal{X}_{a-1}, \mathcal{X}_a), \quad y \in [\mathcal{Y}_{b-1}, \mathcal{Y}_b), \quad v \in [\mathcal{V}_{c-1}, \mathcal{V}_c), \quad t_\Delta = \mathcal{T}_d. \end{aligned} \tag{7}$$

After obtaining the indices (a, b, c, d), we increment A_{abcd}. This is repeated for every triplet within the dataset. The approach is summarized in Algorithm 2. In order to obtain the movement model, i.e., the two-dimensional histogram, we need to condition on a specific speed value v_t as well as a time delta of interest t_Δ and normalize the resulting slice:

$$\begin{aligned}\mathbb{P}_{t_\Delta}\left(\mathbf{p} \mid \mathbf{p}_t, \mathbf{p}_{t-t_\delta}, v_t\right) &= \frac{A_{abcd}}{\sum_{a'=1}^{n_\mathcal{X}} \sum_{b'=1}^{n_\mathcal{Y}} A_{a'b'cd}}, \quad \text{for} \\ (a, b, c, d) &= \text{indices}\left(\psi(\mathbf{p}_{t-t_\delta}, \mathbf{p}_t, \mathbf{p}), v_t, t_\Delta\right). \end{aligned} \tag{8}$$

With this definition, the model can be updated in an online fashion. Moreover, in this way a given position can be evaluated in constant time. If prediction performance is an issue, i.e., for real-time analysis, the counts should be held static. This way, the sums in the denominator can be precomputed leading to an increased computational efficiency.

3.5 Distributed computation

The procedure in Algorithm 2 can be translated to the MapReduce framework (Dean and Ghemawat 2008). Looping over positions in a player's trajectory, we can feed each mapper with a triplet of positions $(\mathbf{p}_s, \mathbf{p}_t, \mathbf{p}_u)$ including the corresponding velocities and a given t_Δ. The mapper transforms the last part of the three-point movement into the space $\mathcal{X} \times \mathcal{Y}$, uses the velocity during the movement and t_Δ and maps the result to the corresponding indices (a, b, c, d). The resulting key-value pair of the mapper consists of the concatenated indices, which serve as the key and a static *one* as the value. We suggest one reducer per matrix entry indexed by (a, b, c, d). Each reducer obtains the concatenated indices as a key and a list of *ones*. The count for A_{abcd} is simply obtained by summing up the ones within the list. The pseudo codes for the mapper and reducer are depicted in Algorithms 3 and 4. After

Algorithm 3 Mapper

```
Input: $(\mathbf{p}_s, \mathbf{p}_t, \mathbf{p}_u)$, $s = t - t_\delta$ and $u = t + t_\Delta$
Output: key $(a, b, c, d)$, value 1
$(a, b, c, d) =$ indices $\left(\psi(\mathbf{p}_s, \mathbf{p}_t, \mathbf{p}_u), v_t, t_\Delta\right)$
```

Algorithm 4 Reducer

```
Input: key $(a, b, c, d)$, values $(1, \ldots, 1)$
Output: value for $A_{abcd}$
return sum of values $(1, \ldots, 1)$
```

executing the MapReduce procedure, we obtain the counts of A for a fixed time delta $t_\Delta = \mathcal{T}_d$. Multiple runs over all time deltas are needed to fill the entire matrix. The movement model is then given as in Eq. (8).

3.6 Complexity

We now consider the complexities for training, prediction, and memory consumption of the proposed approaches for a single player. Let $m_{t_\Delta,v} \in \mathbb{N}$ denote the number of samples within the set $\mathcal{S}_{t_\Delta,V}$ of transformed locations generated by Algorithm 1 when conditioning on a specific time delta t_Δ and speed v, i.e., $m_{t_\Delta,v} = |\mathcal{S}_{t_\Delta,V}|$. First, we consider movement models based on kernel density estimates as introduced in Eq. (6). The complexity of training a KDE is equivalent to the cardinality of the set $\mathcal{S}_{t_\Delta,V}$ and thus equal to $\mathcal{O}(m_{t_\Delta,v})$. Since there is a separate KDE for every time delta and speed, the complexity of training all KDEs for a single player is $\mathcal{O}(\sum_{v\in\tilde{\mathcal{V}}}\sum_{t_\Delta\in\tilde{\mathcal{T}}} m_{t_\Delta,v})$ and thus linear in the player's trajectory data. The complexity of predicting, i.e., obtaining the probability for a given position, using a KDE is $\mathcal{O}(m_{t_\Delta,v})$. Clearly it holds that the larger the training set, the better the model. However, increasing the size of samples $m_{t_\Delta,v}$ makes it prohibitive to use the individual movement models based on KDEs in real-time scenarios. Considering the memory demand of the KDE-based approach, it becomes obvious that all samples are needed as the KDE is a non-parametric method. Hence, $\mathcal{O}\left(\sum_{v\in\tilde{\mathcal{V}}}\sum_{t_\Delta\in\tilde{\mathcal{T}}} m_{t_\Delta,v}\right)$ points need to be stored.

Second, we report on the complexities regarding the count-based movement model as introduced in Eq. (8). The learning procedure is outlined in Algorithm 2 and has the same complexity of constructing all sets $\mathcal{S}_{t_\Delta,V}$ and that is $\mathcal{O}\left(\sum_{v\in\tilde{\mathcal{V}}}\sum_{t_\Delta\in\tilde{\mathcal{T}}} m_{t_\Delta,v}\right)$. Hence, the complexity of building the KDE-based model and the count matrix A is identical. Predicting using the count matrix A when conditioning on a time delta t_Δ and a speed value v is $\mathcal{O}(1)$, assuming no online training. This holds true if the denominator in Eq. (8) is applied not at prediction time but right after the learning. Computing the probability of attaining a given position then boils down to a simple look up within a table. The memory demand of the count matrix A is $\mathcal{O}(n_\mathcal{X} \cdot n_\mathcal{Y} \cdot n_\mathcal{V} \cdot n_\mathcal{T})$. This means that the finer the discretization, the larger the matrix and hence more memory is needed. The complexities of both approaches are summarized in Table 1. It shows that training complexities are equal in both cases. However, evaluating the probability of a position (and thus the most probable location

Table 1 Complexity overview

Approach	Training	Predicting	Memory
KDE-based	$\mathcal{O}\left(\sum_{v\in\tilde{\mathcal{V}}}\sum_{t_\Delta\in\tilde{\mathcal{T}}} m_{t_\Delta,v}\right)$	$\mathcal{O}(m_{t_\Delta,v})$	$\mathcal{O}\left(\sum_{v\in\tilde{\mathcal{V}}}\sum_{t_\Delta\in\tilde{\mathcal{T}}} m_{t_\Delta,v}\right)$
count-based	$\mathcal{O}\left(\sum_{v\in\tilde{\mathcal{V}}}\sum_{t_\Delta\in\tilde{\mathcal{T}}} m_{t_\Delta,v}\right)$	$\mathcal{O}(1)$	$\mathcal{O}(n_\mathcal{X} \cdot n_\mathcal{Y} \cdot n_\mathcal{V} \cdot n_\mathcal{T})$

of a player) is constant for the counting-based approach while it depends on the number of samples for the KDE-based approach; hence, a trade-off between accuracy and speed is possible. Furthermore, the memory requirements grow for the KDE-based approach as more samples are collected. By contrast, the space demand for the counting-based approach is constant in the number of samples but large when high precision and accuracy is necessary.

3.7 Empirical results

There are two typical ways of collecting positional data in sports. The first way is to attach sensors to players and ball to monitor their positions (Grün et al. 2011; Mutschler et al. 2013). The second way is to use computer vision algorithms for retrieving players' and ball's trajectories in consecutive frames (Barris and Button 2008; D'Orazio and Leo 2010). The positional data we use in the experiments stem from the latter and is recorded at 25 Hz. For a single match, this usually yields over $25 \cdot 60 \cdot 90 = 1{,}35{,}000$ samples (due to possible extra time by the end of each half). The dimensions of a soccer field are 105.0 and 68.0 m and the coordinates of positions in the trajectory data are given relative to the origin of the field, which is set to (0, 0). Hence, player coordinates (x, y) are within $F = [-52.5, +52.5] \times [-34.0, +34.0] \subset \mathbb{R}^2$.

Except for the Voronoi-based approach, the models discussed in Sect. 3 involve user-defined parameters that need to be specified. For Taki & Hasegawa, the acceleration parameter $a_{\max}$ can be derived from the corresponding speed samples v_t using $a_t = \frac{1}{h}(v_{t+h} - v_t)$. Here, these are computed for a time horizon of $h = 1$ s using data from a single match. Based on this, we set $a_{\max} = 4.2\,\mathrm{m/s}^2$, which is equal to the 0.999-quantile of the derived values. The quantile instead of the maximum acceleration observed is used to ignore outliers. The model by Fujimura & Sugihara includes two parameters, α and $v_{\max}$. We use $\alpha = 1.3$, which is the value proposed in the original paper (Fujimura and Sugihara 2005), and $v_{\max} = 8.0\,\mathrm{m/s}$, where the latter corresponds to the 0.999-quantile of the observed speed values (analogously as in the case of $a_{\max}$ parameter in the previous model).

To compute the individual movement models presented in Sect. 3.3, we use $t_\delta = 0.2$ s and $t_\Delta = 1$ s in Algorithm 1. We use five different speed intervals shown in Table 2. Note that such a discretization is a common way to bin velocities to account for sparseness in real data, as the number of samples per speed interval may vary significantly (Lago-Peñas et al. 2009; Coutts et al. 2010; Gudmundsson and Wolle 2014). Table 2 also presents speed distributions for three different players: a goalkeeper, a defender, and an attacking midfielder. On average, field players walk and jog and save their energy for only a few sprints.

Movement estimates for these three players are presented in Fig. 4 using a Gaussian KDE with bandwidth equal to 1.0 for simplicity. Note that there are small but distinctive differences between players' ability to move.[3] For example, the goalkeeper has a significantly lower probability to reach distant positions compared to the field players. The reason lies, however, not in her ability to move but in the lack of corresponding observations: Goalkeepers hardly push forward and usually cover a smaller radius than field players. The figures clearly show that the midfielder covers a wider area and is, on average, moving faster than her peers. The few data samples collected for the goalkeeper could be balanced with an average model, see discussion in Sect. 5.

[3] Differences in Table 2 between the defender and the midfielder are significant according to a χ^2-test.

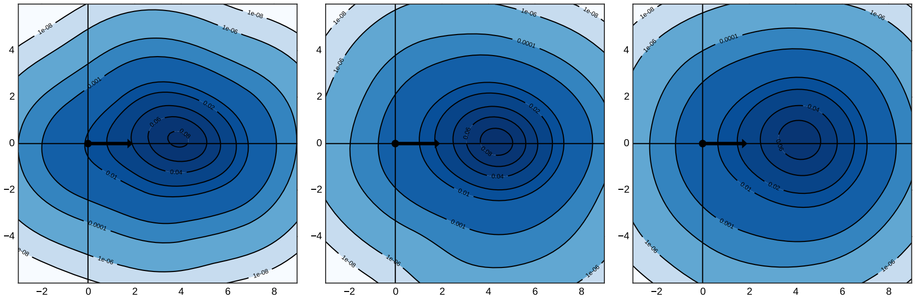

Fig. 4 Individual movement models for three different players with initial speed in the range 14–20 km/h in the direction of the x-axis: goalkeeper (left), defender (center), and midfielder (right)

Table 2 Distribution of speed classes for three different players

Speed	Range (km/h)	Goalkeeper (%)	Defender (%)	Midfielder (%)
Stand	< 1	27.19	11.25	12.01
Walk	$1-7$	66.68	53.22	50.90
Jog	$7-14$	5.44	27.57	28.11
Run	$14-20$	0.57	5.97	6.36
Sprint	> 20	0.12	2.00	2.62

4 Zones of control

4.1 Motivation

Movement models can be used to compute zones of control (or dominant regions) of individual players and teams as a whole (Taki and Hasegawa 2000; Gudmundsson and Wolle 2014; Horton et al. 2015). Below we formally define dominant regions for the models presented in the previous section. To do so, it is beneficial to recall the definition of the traditional movement models that are inspired by physical laws. The definition of probabilistic models is analogous and discussed later.

Let function $\Gamma : \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$ yield the time needed to reach position $\mathbf{p} \in \mathbb{R}^2$ for a player k at position $\mathbf{p}_t^k$ moving with velocity v_t^k in a given direction, i.e., $\Gamma\left(\mathbf{p} \mid \mathbf{p}_t^k, v_t^k\right)$. This function is specific to a given physical model governing player movements. In other words, for a given player, function Γ yields the minimal time that satisfies Eq. (1) and (2) for the Taki & Hasegawa and the Fujimura & Sugihara models, respectively. In Taki and Hasegawa (2000), the concept of a player's zone of control is defined as follows.

Definition 1 The *zone of control* of player i is defined as the subset D^i of the playing area field F, where player i can arrive before any other player $k \neq i$.

Formally, this is to say that $D^i \subseteq F$ is defined such that $\forall \mathbf{p} \in D^i$:

$$i = \arg\min_{k \in \{1,2,\ldots,K\}} \Gamma\left(\mathbf{p} \mid \mathbf{p}_t^k, v_t^k\right).$$

It should be noted that interdependencies between players may be complex enough to produce a player's zone of control that is not a single connected region (Taki and Hasegawa 2000).

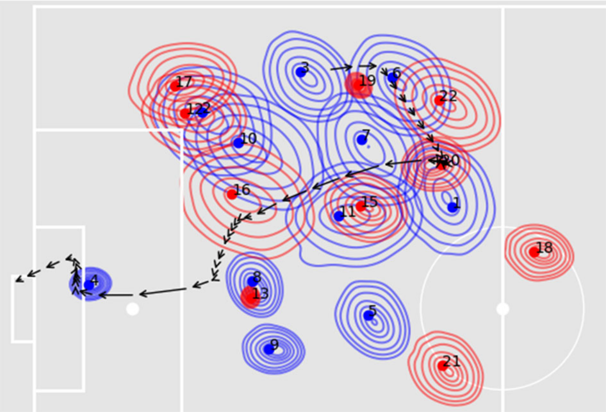

Fig. 5 The same scene as in Fig. 1 but with all movement models for all players shown

The zone of control of a team is defined analogously. Note that in our setup a different perspective is taken by considering probabilistic movement models for a given time horizon. That is, the zones of control are derived on the basis of density functions of possible players' whereabouts. Therefore, we obtain probability distributions of individual players over the playing area. This is depicted in Fig. 5. The computation of those regions using probabilistic movement models is presented in detail below.

4.2 Problem formulation

Let $\mathbb{P}^k_{t_\Delta}\big(\mathbf{p} \mid \mathbf{p}_t, \mathbf{p}_{t-t_\delta}, v_t\big)$ be the movement model of the kth player as introduced in Eq. (6). It quantifies the likelihood of player k to reach position $\mathbf{p}$ given her current $\mathbf{p}_t$ and last position $\mathbf{p}_{t-t_\delta}$, velocity v^k_t, and time horizon t_Δ. The position $\mathbf{p}$ is controlled by player i having the highest likelihood:

$$i = \operatorname*{arg\,max}_{k \in \{1,2,\ldots,K\}} \mathbb{P}^k_{t_\Delta}\big(\mathbf{p} \mid \mathbf{p}_t, \mathbf{p}_{t-t_\delta}, v_t\big).$$

Hence, we can define a function

$$\phi_{t_\Delta} : F \to \{1, 2, \ldots, K\}, \quad \mathbf{p} \mapsto \operatorname{arg\,max}_{k \in \{1,2,\ldots,K\}} \mathbb{P}^k_{t_\Delta}\big(\mathbf{p} \mid \mathbf{p}_t, \mathbf{p}_{t-t_\delta}, v_t\big)$$

that determines the index of the dominating player. Thus, the zone of control of a player i is given as the set of all points $D^i = \{\mathbf{p} \in F \mid \phi_{t_\Delta}(\mathbf{p}) = i\}$ that are controlled by her. It should be noted that ties may occur if the likelihood of two or more players is equal, especially in the counting-based setting. If ties are broken, then the set $\{D^1, D^2, \ldots, D^K\}$ is a partition of F. The procedure is summarized in Algorithm 5.

4.3 Approximating zones of control

Unfortunately, the execution of Algorithm 5 is not practicable. This is because the set F is not iterable since it is uncountable. A typical workaround is to use a finite approximation of the playing area (Nakanishi et al. 2009; Lucey et al. 2012; Narizuka et al. 2014; Franks et al. 2015). Let $G \subset F$ be a finite grid over F containing $n_x \cdot n_y$ equally spaced points in F with (axis-aligned) distance Δ to each other. Player domination is then computed using G rather than F, which yields a finite approximation of the zones of control with precision Δ.

Algorithm 5 Exact computation

```
Input: Movement models $\mathbb{P}^k_{t_\Delta}$ for players $k = 1, 2, \dots, K$
Output: Sets $D^1, D^2, \dots, D^K$
for $k = 1, 2, \dots, K$ do
    $D^k = \{\mathbf{p} \in F \mid \phi_{t_\Delta}(\mathbf{p}) = k\}$
end for
```

Algorithm 6 Finite approximation

```
Input: Movement models $\mathbb{P}^k_{t_\Delta}$ for players $k = 1, 2, \dots, K$
Output: Set $B$
$B = \emptyset$
for $\mathbf{g} \in G$ do
    $B = B \cup \{(\mathbf{g}, \phi_{t_\Delta}(\mathbf{g}))\}$
end for
```

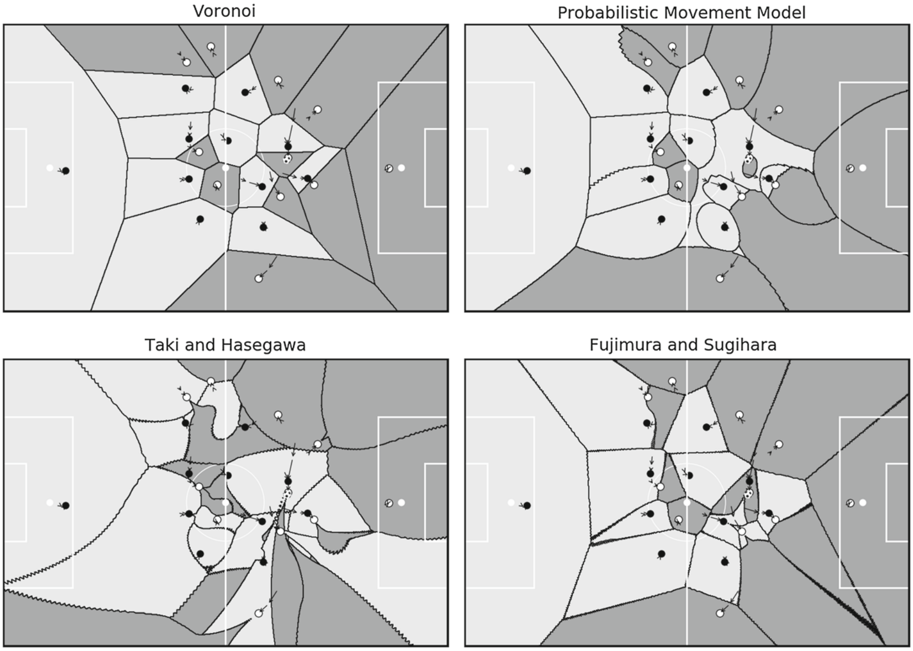

Fig. 6 Controlled zones for different movement models. Black plays from left to right. The two arrows attached to player positions indicate their whereabouts one and two seconds ago, respectively

The smaller Δ is, the better the approximation. The procedure is presented in Algorithm 6. For visualization purposes, the set $B = \{(\mathbf{g}, \phi_{t_\Delta}(\mathbf{g})) \mid \mathbf{g} \in G\}$ can then be used to compute zones of control by assigning each position $\mathbf{p} \in F$ the same label as its closest neighbor from the grid G.

4.4 Empirical results

We now compare zones of control obtained by a Voronoi tessellation, the movement models by Taki & Hasegawa and Fujimura & Sugihara, respectively, and the proposed data-driven movement model for the same situation. Figure 6 shows the resulting regions where arrows indicate directions and velocities of movements.

The top left shows a Voronoi tessellation and implements the assumption that every player is able to run in any direction equally fast, hence ignoring actually observed movements. In other words: the closest player always wins and borders of controlled zones are half cuts between players. The assumption leads to implausible zones of control as we will showcase on the example of the white team playing right to left. The white player on the right wing, for example, has a large zone although she is running towards the center of the pitch. Most of the controlled area of that player lies in her back and she would need to turn before being able to head in that direction. The Voronoi model clearly overestimates the right wing of the white team. By contrast, their left wing is underestimated. Although the left winger pushes forward and although her direct opponents only move slowly and head towards the center of the pitch, her zone is small. In contrast to Voronoi tessellations, the proposed approach in the upper right part of the figure clearly eliminates the depicted limitations. For the white team, the zone of the right winger is realistically small and the zone of the left winger realistically large.

Computing controlled zones using the movement model by Taki & Hasegawa leads to the bottom left figure. Borders between zones are often curly as a direct consequence of the nested circles that originate from the assumption that players may accelerate in any direction unbounded (see Fig. 2). The zone of the white left winger evolves drop-like from the actual player position. The underlying movement model also assigns a big part of the right half of the pitch to the black team although white players are closer positioned and some of them even move into this direction. Figure 6 exhibits the limitations of the approach by Taki & Hasegawa.

The movement model by Fujimura & Sugihara corrects some of the limitations of the model by Taki & Hasegawa and, correspondingly, the bottom right part of Fig. 6 appears more realistic. For instance, similar to the proposed approach, the zone of the white left winger seems more appropriate than the Voronoi-based zone. Nevertheless, there are other problems with this model as can be seen on the right wing of the white team. The zone of the winger has shrunk to almost zero although her opponent is still far away and both are moving slowly.

To sum up, out of the four movement models, only the proposed approach leads to realistic controlled zones that are in line with player movements and distances. Either of the competitors suffer from oversimplified assumptions in the movement models and yield unrealistic zones of control. Analyses that build upon one of the three competitors are likely to be crude as they rely on rough approximations of reality. We include more examples of the methods in the “Appendix”.

5 Discussion

The previous sections show theoretically and empirically that existing movement models suffer from implausible assumptions. Particularly in the previous section, we observe the clear influence of such oversimplifications in the resulting zones of control for Voronoi tessellations and underlying movement models by Taki & Hasegawa and Fujimura & Sugihara.

The idea of this paper is to avoid cumbersome definitions of complex physics (and possibly oversimplifications) by simply observing player movements. We propose a purely data-driven movement model that intelligently combines all movements of a player into either a probabilistic model or grid-based frequencies. Depending on the application at hand, either the full distribution, some quantile thereof, or the convex hull of observed positions can be processed to compute reachable positions in a predetermined time. Further exploiting the probabilistic nature of the model (or turning the frequencies into probabilities) may provide confidences to possible movements. Empirically, the resulting zones of control are intuitive and can be straightforwardly interpreted with player movements and, hence, constitute a realistic picture of a situation.

As a remark, we note that the zones of control for the three baseline approaches are identical when no player is moving. This can be seen by setting $v = 0$ in Eq. (1) and (2) for Taki & Hasegawa and Fujimura & Sugihara, respectively, which then reduces the resulting zones of control to a Voronoi tessellation. The time needed to reach an arbitrary position is now a strictly increasing function of the distance to that position. As Fig. 2 shows, the greater the velocities of the players, the greater the differences of the resulting zones.

However, also note that using positional data for estimating movements of players also comes with limitations. The angle estimation from trajectory data via Eq. (4), for instance, is based on the assumption that players always move forward. In other words, the model assumes that the direction a player is facing is in line with her movement. This is not always the case as particularly goalkeepers often move backwards. The model would thus over- or underestimate the time needed for turning around depending on the actual change of direction. A possible remedy could be a better approximation of the angle θ rather than using Eq. (4) or devising the angle from an auxiliary data source. Using positional data alone is, however, not sufficient to solve this matter.

The goalkeeper serves as an example for another problem of the proposed approach as she is hardly running at full speed. Thus, just by observing her movements on the pitch, one will hardly be able to assess her full potential. The same problem occurs with players that are substituted for the first time as the proposed approach does not apply off-the-shelf to unseen players. The problem is also known as the *cold-start problem* and similar instances occur in recommendation scenarios (see, e.g., Son 2016). To overcome this problem, a two-component mixture model can be used. The first component utilizes the actual (and continuously updated) movement model $\mathbb{P}^k_{n_k}$ of the new player k, which is learned on n_k points. The second component is an *average* model $\mathbb{P}^{\text{avg}}$ over all players (with a similar role) and their data points. The idea is to blend the personalized component with the average component until the former is accurate enough to be used alone. Hence, the model is given by a convex combination of movement models

$$\mathbb{P}^k = \lambda \cdot \mathbb{P}^k_{n_k} + (1 - \lambda) \cdot \mathbb{P}^{\text{avg}} \quad \text{with} \quad \lambda = \min\left(\frac{n_k}{N}, 1\right).$$

If $n_k = 0$, then only the average model will be used. Once n_k exceeds the number of data points N, $\lambda = 1$ and the average model is weighted by zero and hence automatically deactivated as desired. The required number of observations depends both on the domain and a player's speed. In case of soccer and for a given speed range, several thousands of samples appear sufficient to produce satisfactory results. For a field player, those samples can, for instance, be collected in a single match. However, in case of a goalkeeper, it is recommended to always maintain an additional average movement model due to the small number of samples for higher values of initial velocities as she mostly stands or walks during a match. Note that this mixture-approach works for both, the movement models based on KDEs and based on the count matrix A.

There are many possible use cases where realistic movement models may give an edge toward existing techniques. For instance, player performance indices that ground on the ability to move (Taki et al. 1996; Ueda et al. 2014) may be revised accordingly. Similarly, player ratings that measure to what extend their controlled zone contributes to the overall area controlled by their team (Link et al. 2016; Harmon et al. 2016) may be revisited. Figure 7 shows a potential application that deals with estimating probabilities of passing and pass completion given the context of the ball possessing player to test the hypothesis that players try more difficult passes when they have enough space. While the space is directly given by their zone of control, pass interception and pass completion probabilities could be conditioned

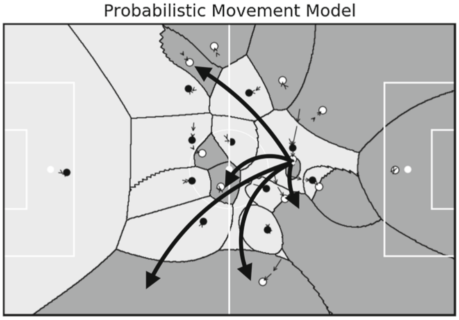

Fig. 7 Mockup showing possible passes

on the available area to shed light on which player to attack in what situations and also where to position the own defenders to possibly intercept and defend the receiving player.

Along these lines is also the prediction of pass outcomes (Nakanishi et al. 2009). The idea is to split the trajectory of the ball into small units that are processed one after another. For every unit, the probability that a player reaches the position of the ball during the lifespan of the unit is computed. If an opposing player fulfills this criterion, she intercepts the ball and the computation terminates. If no player intercepts the ball, the pass is completed after processing the final unit. A preliminary 10-fold cross validation on 1194 passes shows that underlying probabilistic movement models lead to prediction accuracies of 97.5% for pass interception and 88.5% for pass completion. The approaches by Taki & Hasegawa and Fujimura & Sugihara perform similarly and achieve accuracies of about 93.7 and 94.5 % for interception and 69.1 and 79.5 % for completion, respectively. Voronoi tesselations perform worst and yield only a correct interception in 50.9%. Note that the underlying Voronoi model always predicts an interception since there is always a player closer to the ball trajectory than the player making the pass in at least one unit. However, a thorough evaluation is necessary to confirm these promising results.

6 Conclusion

We proposed a novel data-driven method for estimating individual movement models using positional data. The model is generated by conditioning a player's whereabouts after a given time on her initial position and velocity. We obtained tables of reachable (x, y) coordinates for every velocity and time interval and proposed to turn these tables into a probabilistic movement model using kernel density estimation. Alternatively, the tables may be discretized using a grid (ϵ-net) to work directly with counts instead of probabilities and hence speed-up computation for (near) real-time scenarios. Movement models were computed for every player individually and the computation could be distributed on many machines to compute movement models for many players and process many games at once. Empirically, we showed the limitations of existing movement models and exemplified the usefulness of the contribution on the example of zones of control. Computing these zones using existing approaches led to crude approximation due to oversimplified assumptions in the respective models. By contrast, the proposed movement models led to realistic and intuitive zones of control.

Acknowledgements The authors would like to thank Hendrik Weber and Deutsche Fußball Liga (DFL) and Sportcast GmbH for providing positional data.

Appendix

To shed more light on the different movement models and their implication on the zones of control, the following Figures show four exemplary situations for Voronoi-based movement models, approaches by Taki & Hasegawa and Fujimura & Sugihara as well as the proposed probabilistic movement model.

Figure 8 shows perhaps the most relevant situation for coaches and analysts. The probabilistic movement model on the top right clearly identifies a scoring opportunity for the white team. The white player in the center of the pitch creates a large zone of control behind the defending black players. If the ball possessing player plays the ball into this zone, the white player may have enough time to control the ball and to create a one-on-one with the goal keeper. Except for Taki & Hasegawa, the baselines fail to detect this.

In general, Fig. 9 shows that the baselines either lead to unnatural square- and rectangle-like shapes (Voronoi and Fujimura & Sugihara) or implausible drop-like areas (Taki & Hasegawa) as a consequence of implicit assumptions and constraints in the models. Our approach allows to capture movements irrespectively of the resulting shapes of the zones as there are no assumptions on the movements.

Velocities are generally an issue for the baseline approaches. Figure 10 shows an example where we focus only on the ball possessing player and the white striker that runs towards her. The region in the Voronoi-based approach are clearly too small for the running player. By

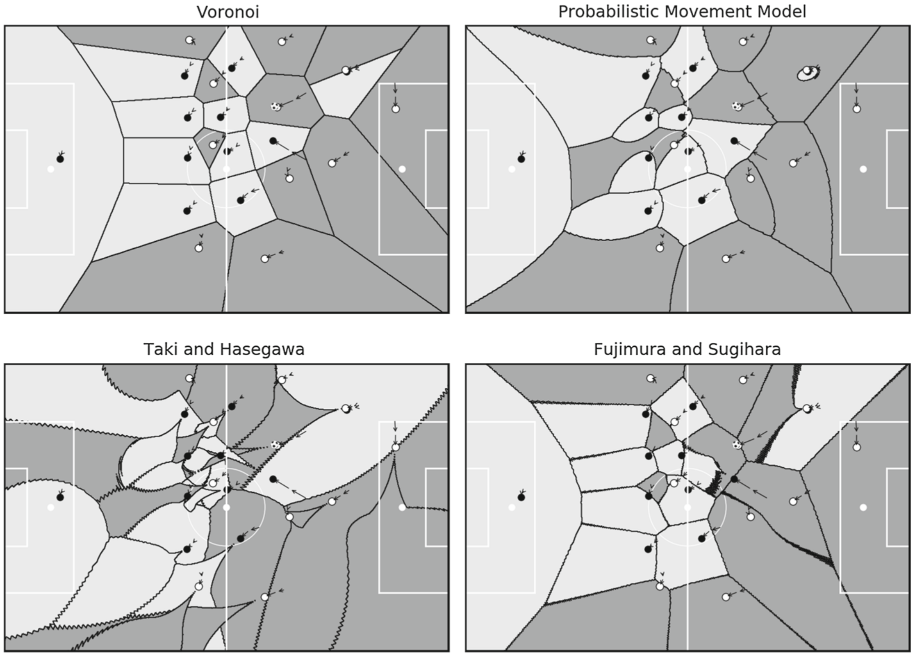

Fig. 8 Scoring opportunity for the white team

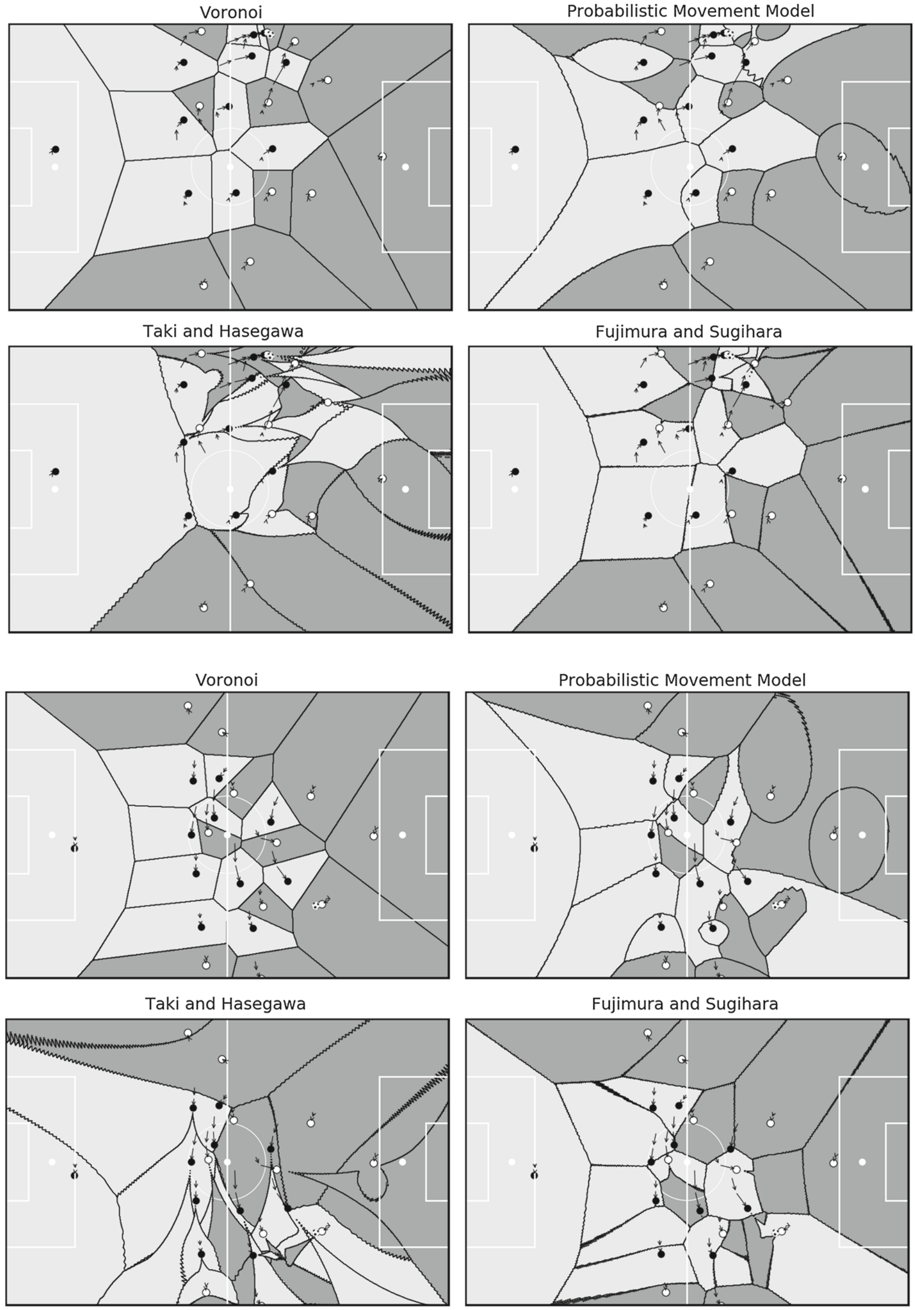

Fig. 9 Implicit assumptions in baselines constrain possible shapes of zones

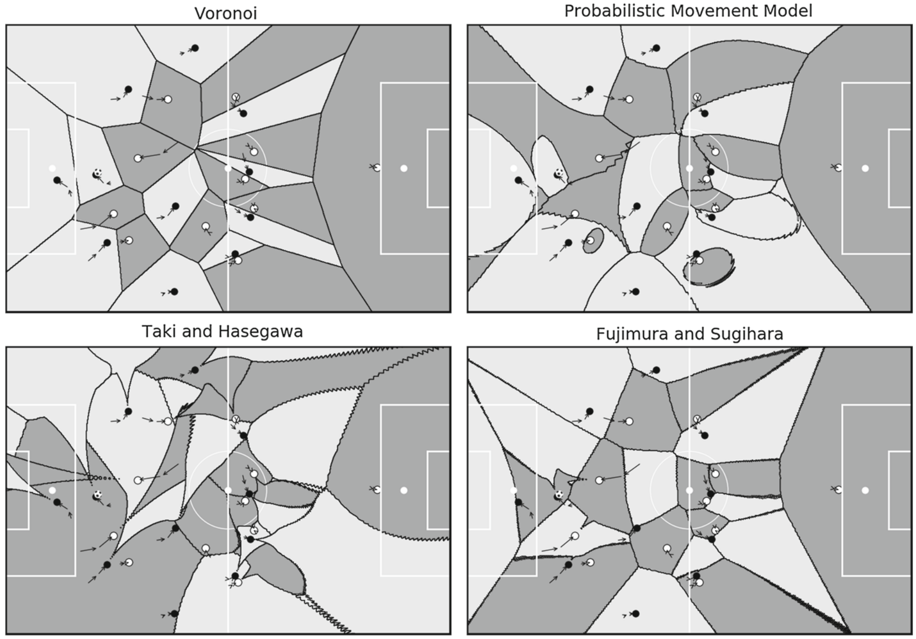

Fig. 10 High velocities not appropriately captured by baselines

contrast, Taki & Hasegawa and Fujimura & Sugihara overestimate the impact of the approaching white player and render the ball possessing player outside of her own region of control. The approach by Taki & Hasegawa even credits a surprisingly large area to the second white player from the left. Interestingly, this player is almost standing and only gets an area this large because the zones of the other players evolve drop-like into the direction of movement. A remedy to such artefacts is to compute the controlled zones with underlying probabilistic movement models. The respective figure on the top right shows realistic areas that are easily interpreted.

References

Avci, A., Bosch, S., Marin-Perianu, M., Marin-Perianu, R., & Havinga, P. (2010). Activity recognition using inertial sensing for healthcare, wellbeing and sports applications: A survey. In *23th International conference on architecture of computing systems 2010*, pp. 1–10.

Barris, S., & Button, C. (2008). A review of vision-based motion analysis in sport. *Sports Medicine*, *38*(12), 1025–1043.

Brooks, J., Kerr, M., & Guttag, J. (2016). Using machine learning to draw inferences from pass location data in soccer. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *9*(5), 338–349.

Byrne, M., Parry, T., Isola, R., & Dawson, A. (2013). Identifying road defect information from smartphones. *Road & Transport Research*, *22*(1), 39–50.

Coutts, A. J., Quinn, J., Hocking, J., Castagna, C., & Rampinini, E. (2010). Match running performance in elite Australian rules football. *Journal of Science and Medicine in Sport*, *13*(5), 543–548.

Dean, J., & Ghemawat, S. (2008). MapReduce: Simplified data processing on large clusters. *Communications of the ACM*, *51*(1), 107–113.

D'Orazio, T., & Leo, M. (2010). A review of vision-based systems for soccer video analysis. *Pattern Recognition*, *43*(8), 2911–2926.

Fonseca, S., Milho, J., Travassos, B., & Araújo, D. (2012). Spatial dynamics of team sports exposed by voronoi diagrams. *Human Movement Science*, *31*(6), 1652–1659.

Franks, A., Miller, A., Bornn, L., & Goldsberry, K. (2015). Characterizing the spatial structure of defensive skill in professional basketball. *The Annals of Applied Statistics*, *9*(1), 94–121.

Fujimura, A., & Sugihara, K. (2005). Geometric analysis and quantitative evaluation of sport teamwork. *Systems and Computers in Japan*, *36*(6), 49–58.

Gottfried, B. (2008). Representing short-term observations of moving objects by a simple visual language. *Journal of Visual Languages & Computing*, *19*(3), 321–342.

Gottfried, B. (2011). Interpreting motion events of pairs of moving objects. *GeoInformatica*, *15*(2), 247–271.

Grün, Tvd, Franke, N., Wolf, D., Witt, N., & Eidloth, A. (2011). *A real-time tracking system for football match and training analysis* (pp. 199–212). Berlin Heidelberg: Springer.

Gudmundsson, J., & Horton, M. (2017). Spatio-temporal analysis of team sports. *ACM Computing Surveys*, *50*(2), 22:1–22:34.

Gudmundsson, J., & Wolle, T. (2014). Football analysis using spatio-temporal tools. *Computers, Environment and Urban Systems*, *47*, 16–27.

Haase, J. & Brefeld, U. (2014). Mining positional data streams. In *International workshop on new frontiers in mining complex patterns*, pp. 102–116. Springer.

Harmon, M., Lucey, P., & Klabjan, D. (2016). Predicting shot making in basketball learnt from adversarial multiagent trajectories. *ArXiv e-prints*.

Horton, M., Gudmundsson, J., Chawla, S., & Estephan, J. (2015). Automated classification of passing in football. In *Pacific-Asia conference on knowledge discovery and data mining*, pp. 319–330. Springer.

Janetzko, H., Sacha, D., Stein, M., Schreck, T., Keim, D. A., & Deussen, O. (2014). Feature-driven visual analytics of soccer data. In *2014 IEEE conference on visual analytics science and technology (VAST)*, pp. 13–22.

Knauf, K., Memmert, D., & Brefeld, U. (2016). Spatio-temporal convolution kernels. *Machine Learning*, *102*(2), 247–273.

Lago-Peñas, C., Rey, E., Lago-Ballesteros, J., Casais, L., & Domínguez, E. (2009). Analysis of work-rate in soccer according to playing positions. *International Journal of Performance Analysis in Sport*, *9*(2), 218–227.

Lasek, J. & Gagolewski, M. (2015). The winning solution to the AAIA'15 data mining competition: Tagging firefighter activities at a fire scene. In *2015 Federated conference on computer science and information systems (FedCSIS)*, pages 375–380.

Laube, P., Imfeld, S., & Weibel, R. (2005). Discovering relative motion patterns in groups of moving point objects. *International Journal of Geographical Information Science*, *19*(6), 639–668.

Le, H. M., Carr, P., Yue, Y., & Lucey, P. (2017). Data-driven ghosting using deep imitation learning. In *MIT sloan sports analytics conference*.

Link, D., Lang, S., & Seidenschwarz, P. (2016). Real time quantification of dangerousity in football using spatiotemporal tracking data. *PLoS ONE*, *11*(12), 1–16.

Lucey, P., Bialkowski, A., Carr, P., Foote, E., & Matthews, I. (2012). Characterizing multi-agent team behavior from partial team tracings: Evidence from the English Premier League. In*Proceedings of the twenty-sixth AAAI conference on artificial intelligence*, AAAI'12, pp. 1387–1393. AAAI Press.

Mazimpaka, J. D., & Timpf, S. (2016). Trajectory data mining: A review of methods and applications. *Journal of Spatial Information Science*, *2016*(13), 61–99.

Memmert, D., Lemmink, K. A. P. M., & Sampaio, J. (2016). Current approaches to tactical performance analyses in soccer using position data. *Sports Medicine*, *47*, 1–10.

Mohan, P., Padmanabhan, V. N., & Ramjee, R. (2008). Nericell: Rich monitoring of road and traffic conditions using mobile smartphones. In *Proceedings of the 6th ACM conference on embedded network sensor systems, SenSys '08*, pp. 323–336. ACM.

Mutschler, C., Ziekow, H., & Jerzak, Z. (2013). The DEBS 2013 grand challenge. In *Proceedings of the 7th ACM international conference on distributed event-based systems, DEBS '13*, pp. 289–294, New York, NY: ACM.

Nakanishi, R., Maeno, J., Murakami, K., & Naruse, T. (2009). An approximate computation of the dominant region diagram for the real-time analysis of group behaviors. In *Robot soccer world cup*, pp. 228–239. Springer.

Narizuka, T., Yamamoto, K., & Yamazaki, Y. (2014). Statistical properties of position-dependent ball-passing networks in football games. *Physica A: Statistical Mechanics and its Applications*, *412*, 157–168.

Paefgen, J., Michahelles, F., & Staake, T. (2011). GPS trajectory feature extraction for driver risk profiling. In *Proceedings of the 2011 international workshop on trajectory data mining and analysis, TDMA '11*, pp. 53–56, New York, NY: ACM.

Rossi, A., Pappalardo, L., Cintia, P., Fernandez, J., Iaia, F. M., & Medina, D. (2017). Who is going to get hurt? Predicting injuries in professional soccer. In *Proceedings the machine learning and data mining for sports analytics workshop (MLSA'17), ECML/PKDD*, CGI '00, pp. 227–235.

Son, L. H. (2016). Dealing with the new user cold-start problem in recommender systems: A comparative review. *Information Systems*, *58*, 87–104.

Sprado, J., & Gottfried, B. (2009). *What motion patterns tell ss about soccer teams* (pp. 614–625). Heidelberg: Springer.

Taki, T.& Hasegawa, J. (2000). Visualization of dominant region in team games and its application to teamwork analysis. In *Proceedings of the international conference on computer graphics, CGI '00*, pp. 227–235, Washington, DC: IEEE Computer Society.

Taki, T., Hasegawa, J., & Fukumura, T. (1996). Development of motion analysis system for quantitative evaluation of teamwork in soccer games. In *Proceedings of 3rd IEEE international conference on image processing*, vol. 3, pp. 815–818.

Turlach, B. A. (1993). Bandwidth selection in kernel density estimation: A review. In *CORE and institut de statistique*.

Ueda, F., Masaaki, H., & Hiroyuki, H. (2014). The causal relationship between dominant region and offense-defense performance—Focusing on the time of ball acquisition. *Football Science*, *11*, 1–17.

Voronoi, G. (1908). Nouvelles applications des paramètres continus à la théorie des formes quadratiques. premier mémoire. Sur quelques propriétés des formes quadratiques positives parfaites. *Journal für die reine und angewandte Mathematik*, *133*, 97–178.

Zhang, P., Beernaerts, J., Zhang, L., & de Weghe, N. V. (2016). Visual exploration of match performance based on football movement data using the continuous triangular model. *Applied Geography*, *76*(Supplement C), 1–13.

Zhao, Y., Yin, F., Gunnarsson, F., Hultkratz, F., & Fagerlind, J. (2016). Gaussian processes for flow modeling and prediction of positioned trajectories evaluated with sports data. In *2016 19th international conference on information fusion (FUSION)*, pp. 1461–1468.

Zheng, S., Yue, Y., & Hobbs, J. (2016). Generating long-term trajectories using deep hierarchical networks. *In Advances in Neural Information Processing Systems*, *29*, 1543–1551.

Zheng, Y. (2015). Trajectory data mining: An overview. *ACM Transactions on Intelligent Systems and Technology*, *6*(3), 29:1–29:41.