Unifying Microscopic Flocking Motion Models for Virtual, Robotic, and Biological Flock Members

Benjamin T. Fine and Dylan A. Shell
Department of Computer Science and Engineering
Texas A&M University

May 9, 2013

Abstract

Flocking motions have been the subject of hundreds of studies over the past six decades. The vast majority of models have nearly identical aims: bottom-up demonstration of basic emergent flocking motions. Despite a significant fraction of the literature providing algorithmic descriptions of models, incompleteness and imprecision are also readily identifiable in flocking algorithms, algorithmic input, and validation of the models. To address this issue, this meta-study introduces a data-flow template, which unifies many of the existing approaches. Additionally, there are small differences and ambiguities in the flocking scenarios being studied by different researchers; unfortunately, these differences are of considerable significance. For example, much subtlety is needed to specify sensory requirements exactly and minor modifications may critically alter a flock's exhibited motions. We introduce two taxonomies that minimize both incompleteness and imprecision, and enable us to highlight those publications that study flocking motions under comparable assumptions. Furthermore, we aggregate and translate the publications into a consolidated notation. The common notation along with the data-flow template and the two taxonomies constitute a collection of tools, that together, facilitates complete and precise flocking motion models, and enables much of the work to be unified. To conclude, we make recommendations for more diverse research directions and propose criteria for rigorous problem definitions and descriptions of future flocking motion models.

1 Introduction

Flocking behaviors by groups of individuals has been extensively studied for the past six decades in multiple research communities, including biology (Aoki, 1984; Hildenbrandt et al, 2010; Vine, 1971), physics (Vicsek et al, 1995; Szabó et al, 2008; Czirók et al, 1997), and robotics (Arkin and Balch, 1999; Gazi and Passino, 2005; Ferrante et al, 2012). Although many works have expressed confidence in our comprehension of this phenomenon (Goldstone and Janssen, 2005), understanding of the flocking phenomenon remains incomplete (Giardina, 2008; Parrish et al, 2002; Vicsek and Zafeiris, 2012; Lopez et al, 2012; Hildenbrandt et al, 2010). While several candidate models are available (*e.g.*, Wood and Ackland (2007); Warburton and Lazarus (1991); Rauch et al (1995); Okubo (1986); Moussaïd et al (2009); James et al (2004); Helbing et al (2000); Hauert et al (2011)), there is no consensus on the precise details of the motions needed to produce rich flocking motions under realistic sensing models, actuation, and dynamics constraints. This stems, we believe, partly from a poor definition of what constitutes a flocking behavior.

Although diverse research communities study different varieties of the problem and questions surrounding the phenomenon (Clark and Evans, 1954; Dingle and Drake, 2007; Emlen, Jr., 1952; Partridge, 1982; Pitcher et al, 1976; Rands et al, 2004; Viscido and Wethey, 2002; Whitfield, 2003; Simons, 2004; Parrish and Edelstein-Keshet, 1999; Parrish, 1989; Miki and Nakamura, 2006; Hutto, 1988; Edelstein-Keshet, 2001; Bender and Fenton, 1970), the vast majority of the flocking literature aims at bottom-up production of flocking motion (Goldstone and Janssen, 2005; Parrish et al, 2002). Generally, studies are reported without explicitly

detailing the sensing capabilities, limiting assumptions, and/or computation capabilities of the individual flock members. Additionally, there is currently no common or accepted method for the design, validation, and/or presentation of flocking motion models (Parrish et al, 2002; Vicsek and Zafeiris, 2012), which makes it difficult to determine the current state of the literature and to compare existing motion models.

We have reviewed over one-hundred publications from various communities and have selected a subset of thirty-two to conduct a detailed meta-study. These publications were carefully chosen to maximize coverage of the common design choices and assumptions found throughout the literature and to be a representative cross-section of the literature as a whole. We have chosen to included one of our recent publications (Fine and Shell, 2011) that was developed with the conclusions of this work in mind. In Section 6 we will use that study as a *case-study* to highlight the use of the tools presented here.

Our study reveals that the current presentation of proposed motion models lack either completeness, precision, or both, which significantly hinders repeatability. Additionally, there are small, sometimes subtle, implicit and/or explicit assumptions that are currently overlooked (examples we have teased out include: member-based detection (Fine and Shell, 2011, 2012), lack of occlusions (Fine and Shell, 2011), perfect sensing (Fine and Shell, 2011)), which may impact the produced motions. Furthermore, even models that are completely and precisely presented are not always realized exactly when validated (*e.g.*, a motion model designed for local sensing might be validated using global sensing). Therefore, it is difficult to know the degree to which a particular motion model is actually capable of producing flocking motion in a realistic scenario or if the model's assumptions are realistic. To clarify, we use the following terms when referring to flocking motion models:

- **Completeness** refers to how many of the *key* aspects of the flocking model (Section 2) are presented.

- **Precision** (or lack thereof) is the quality of the specification/presentation of the various aspects of the flocking motion model.

To better understand all aspects involved in the generation of flocking motions, we introduce a series of analytical tools that help reduce incompleteness and imprecision in the design and presentation of these models. Together, these three distinct tools use three different approaches for the understanding of the flocking phenomena; (1) organizational, (2) categorical, and (3) structural. To gain a complete and precise understanding of a given model, we must use all three tools, as they highlight different aspects of the model.

The first tool, the data-flow template (DT), identifies and addresses the key aspects for the production of flocking motions and how these aspects relate. Together, the five aspects, or stages, of the DT (see Section 2) give an organizational description for the production of flocking motions that can be used for understanding what information is required and what stage(s) utilize it. Additionally, the DT can be a useful *blueprint* for the design of new flocking motion models.

Moving beyond the organization of each particular motion model, we sought to chart the relationships across existing models with our second tool; the objective is to categorize existing work concisely while retaining sufficient precision to allow a practitioner to resolve implementation questions. We developed a taxonomy to detail the computation, sensing, and motion capabilities of the individual flock members. Additionally, we introduce a second taxonomy that aids in the classification of validation methods used for a particular flocking motion model. When using the two taxonomies together, one may gain insight into which assumptions or capabilities may be infeasible or impractical for a robotic or biological flock member. Using the two taxonomies together also affords the ability to identify which motion model design choices have been fully validated (*e.g.*, local sensing versus global sensing).

Flocking motion models that appear to be similar at both the organizational and categorical levels may actually differ significantly in the structure of the motion computation. The third tool used for gaining a fuller understanding of the model is a consolidated notation and formalization which focuses on the motion computation of a given motion model (*i.e.*, the motion rule). The motion rule is a combination of the neighbor selection and the motion computation stages and can be considered the *algorithm* of the motion model. Formalization of the current motion rules facilitates understanding the implementation differences between the various motion models. Even though many of the motion models have the same aim, the

formalization and notation shows that the implementation of the models are typically quite different, which can lead to different modeling assumptions.

1.1 Scope

Of the publications reviewed for this study, we only considered literature which uses a microscopic motion model for investigating flocking motions. Microscopic flocking motion models have been the primary focus of the many diverse research communities, thus resulting in several reviews and surveys of the literature (Vicsek and Zafeiris, 2012; Blomqvist et al, 2012; Goldstone and Janssen, 2005; Giardina, 2008; Parrish et al, 2002). These models typically focus on the motion rule used to produce the motions of the individual flock members. Such models have been used to explore why flocks exist (Hamilton, 1971; Partridge, 1982; Viscido et al, 2002; Viscido and Wethey, 2002; Bazazi et al, 2008; Barbosa, 1995), what is required for the production of flocking motions (Reynolds, 1987; Pitcher et al, 1976; Fine and Shell, 2011, 2012), how much influence individuals have on the group (Couzin et al, 2005; Conradt et al, 2009; Warburton and Lazarus, 1991), what special/unique properties might exist in a flock (Vicsek et al, 1995), and how can we apply these motions to robots (Turgut et al, 2008; Arkin and Balch, 1999; Ferrante et al, 2012).

In addition to investigating flocking motion using microscopic models, some work uses macroscopic models (Vaughan et al, 2000; Mogilner and Edelstein-Keshet, 1999; Babak et al, 2004). Instead of studying the individual flock members, macroscopic models typically focus on group-level motions. In some of the literature, macroscopic models are used as a tool for the validation of microscopic models and are not used as a *stand alone* model (Albi and Pareschi, 2012; Cavagna et al, 2012). Such work is considered outside the scope of this paper.

1.2 Notation

To improve precision in understanding the literature, we use the notation presented in Table 1. The chosen notation represents something of a unified superset of notation seen in the selected publications. This notation allows for unambiguous formalization of the selected publications and is used throughout this meta-study.

2 Data-flow Template

The data-flow template (DT) aids in designing and presenting complete microscopic flocking motion models. Each of the five stages (sensing, flock member detection, neighbor selection, motion computation, and physical motion) of the DT represent the key aspects for the generation of flocking motions. The five stages of the DT are connected by the information that is passed between them. With an explicit understanding of the five stages and the relative connections, one gains an complete understanding of the motion model and it facilitates repeatability among researchers. The DT differs from the other tools presented in this work, in that the DT defines microscopic motion models at the logical level. In this way, the DT is a *blueprint* for flocking motion models, as it details the major building blocks and how they fit together.

In addition to serving as a *blueprint* for the creation of flocking motion models, the DT is a useful tool for understanding the complexity of a given model. Complexity, here, refers to how much computation, in the motion computation stage, is required to execute a given model. Therefore, a motion model that uses all of the detected flock members is less complex than a model that only uses a subset of the detected flock members, since this model performs extra computation to produce the subset. The DT used in this way can be useful for gaining a better understanding for which stages can be executed at the hardware level, thus making the model less complex. Figure 1 shows a generic view of the DT along with the connections between the various stages.

<div align="center">**Notation and Definitions**</div>

Bold capital roman letter	Denotes a set (*e.g.*, $\mathbf{A}(t)$).
Bold lowercase roman letter	Denotes a vector (*e.g.*, $\mathbf{v}_i(t)$).
$\doteq$	Definition (*i.e.*, $x \doteq y$ should be read as "x is defined as y").
$*$	Denotes a preference (*e.g.*, $\mathbf{d}_{i*}(t) \doteq$ the direction vector from agent i to location $*$ at time t).
$\hat{\mathbf{x}}_i(t)$	Denotes the normalized form of $\mathbf{x}_i(t)$.
$\mathbf{x}_{i^x}(t)$	Denotes the x component of $\mathbf{x}_i(t)$ (We only use this notation for the first three components; x, y, and z).
$\perp(\mathbf{x}_i(t))$	Denotes the vector perpendicular to $\mathbf{x}_i(t)$.
$\angle(\mathbf{x}_i(t))$	Denotes the argument of the vector $\mathbf{x}_i(t)$ (*i.e.*, the angle describing the direction of a vector).
$\mathbf{A}(t)$	The set of all flock members at time t.
$\mathbf{D}_i(t)$	The set of all flock members detected by the ith flock member at time t (*i.e.*, $\mathbf{D}_i(t) \subseteq \mathbf{A}(t)$).
$\mathbf{I}_i(t)$	The set of all flock members selected by the perception function (*i.e.*, $\mathbf{I}_i(t) \subseteq \mathbf{D}_i(t)$).
$\mathbf{r}_i(t)$	The position of the ith flock member at time t.
$\mathbf{d}_{ij}(t)$	The direction vector from the ith flock member to the jth flock member at time t.
$\mathbf{v}_i(t)$	The velocity of the ith flock member at time t.
$\mathbf{u}_i(t)$	The speed of the ith flock member at time t.
$\theta_i(t)$	The orientation of the ith flock member at time t (w.r.t. some true/global direction).
$\mathbf{R}_{[\alpha,\beta]}$	The set of all valid distances (*i.e.*, any flock member which has a distance $d \in \mathbf{R}_{[\alpha,\beta]}$ would exists in $\mathbf{R}_{[\alpha,\beta]}$).
$L^2(\mathbf{r}_i, \mathbf{r}_j)$	The Euclidean norm between $\mathbf{r}_i(t)$ and $\mathbf{r}_j(t)$ (*i.e.*, $\|\cdot\|_2$).
w_k^i	The kth gain/constant for the ith flock member (if the superscript is omitted then it is a global gain/constant).

Table 1: Definitions and style guidelines for the formalization and unification of the motion models presented in this meta-study.
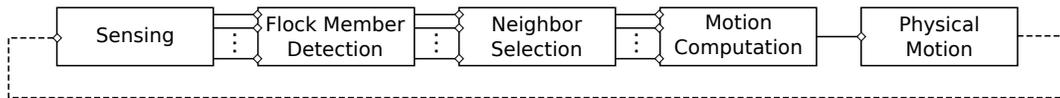
Figure 1: A diagrammatic representation of the proposed DT for microscopic flocking motion models. It details the main aspects for the generation of flocking motions via the five boxes (stages). The connections between the stages encode the data that propagates between them. In particular, the connections between the sensing stage and flock member detection stage represents the raw sensor information from each sensor (*e.g.*, laser range-finder, camera, GPS). The connection between the flock member detection and neighbor selection stage is the set of detected flock members ($\mathbf{D}_i(t)$). The neighbor selection stage passes at least one set of selected flock members ($\mathbf{I}_i(t)$) to the motion computation stage which passes the next computed motion to the physical motion stage.

2.1 The Five Stages: Definitions

2.1.1 Sensing

The sensing stage translates the visible environment (from the individual sensors reference frame* into usable input for the later stages (*e.g.*, a laser range-finder converts the visible environment into a list of ranges). A

---

*Typically the sensor's reference frame will be that of the flock member, but when the model uses global sensors (*e.g.*, overhead camera) the two reference frames will differ.

flock member's internal representation of the visible environment is based on the sensors used, therefore the design of the following stages is directly affected by this stage. For example, if a flock member is equipped with a laser range-finder the flock member detection stage may use shape (based on the type of raw sensor data) to detect other flock members in the environment. Although there is no formal input to the sensing stage, the DT in Figure 1 shows input to the sensing stage from the physical motion stage because the resulting motion of that stage may affect the visible environment.

2.1.2 Flock Member Detection

The flock member detection stage uses the raw sensor information provided by the sensing stage and outputs the set of all detected flock members ($\mathbf{D}_i(t)$). The set $\mathbf{D}_i(t)$ is a subset of all possible flock members, represented by the set $\mathbf{A}(t)$, that are within the visible sensing region ($\mathbf{R}_{[\alpha, \beta]}$). In other words, if the senors defined in the sensing stage only senses information within a two meter radius ($\mathbf{R}_{[0, \ 2 \ \text{meters}]}$), then the set $\mathbf{D}_i(t)$ will only contain flock members that are within a two meter radius. Additionally, each flock member in the set $\mathbf{D}_i(t)$ encompasses all of the required information (*e.g.*, position and velocity) and we assume the type of information used for the description of a flock member has no effect on the DT. Therefore, if the model requires position and orientation information, then for each flock member in $\mathbf{D}_i(t)$, there will be a corresponding $\mathbf{r}_j(t)$ and $\theta_j(t)$.

2.1.3 Neighbor Selection

The neighbor selection stage takes the set $\mathbf{D}_i(t)$ provided by the flock member detection stage and outputs at least one subset of the set $\mathbf{D}_i(t)$. The set(s) generated by the neighbor selection stage only contain flock members which will be used in the motion computation stage. Therefore, if the motion computation stage only uses the nearest neighbor[†] (in distance) to compute the next motion, then the neighbor selection stage will only output a set that contains the nearest neighbor.

To reduce the set $\mathbf{D}_i(t)$ to the desired output, the neighbor selection stage uses a set of perception functions (see Table 2 for a list of perception function definitions). The neighbor selection function can use any number and combination of these functions in order to reduce the set $\mathbf{D}_i(t)$ into usable input for the motion computation stage. The perception functions could be used in succession (*e.g.*, union of the output of two perception functions) or in parallel (*i.e.*, the neighbor selection stage would output more than one set). For example, if the motion computation stage may require two sets as input (*e.g.*, attraction and repulsion sets), the neighbor selection stage will output two sets; the set of 'attraction-zone' flock members and the set of 'repulsion-zone' flock members. One possible representation of the neighbor selection function that considers the attraction-repulsion zones, using the notation in Table 1, is ($\mathbf{I}_{Repulsion}(t) \ \cap \ \mathbf{I}_{Attraction}(t)) \ \subseteq \ \mathbf{D}_i(t)$.

2.1.4 Motion Computation

The motion computation stage uses the set(s) generated by the neighbor selection stage to calculate the next motion of the flock member (*e.g.* this stage can update any combination of $\mathbf{r}_i(t)$, $\theta_i(t)$, $\mathbf{u}_i(t)$, or $\mathbf{v}_i(t)$). It is important to note that this stage only describes the internal representation of the next motion and does not describe how the internal representation is translated into low-level control commands for the flock member. In the case of the attraction/repulsion zone example, the motion computation stage may compute two different motion vectors that are then summed together to produce the next motion. Table 4 describes most of the neighbor selection and motion computation stages in the selected publications.

2.1.5 Physical Motion

The physical motion stage takes the computed motion from the motion computation stage and translates it into a form that can be realized in either a simulated or physical robot (*e.g.*, a kinematic motion model

---

[†]In this paper, flock member X calls member Y a *neighbor* if and only if Y can be sensed by X.

$\text{All}() = \{j \in \mathbf{A} : j\} = \mathbf{A}$

All flock members are selected.

$\text{FixedSet}_i() = \{j \in \mathbf{S}_i \subseteq \mathbf{D}_i : j\} = \mathbf{S}_i$
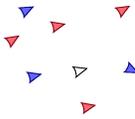
Each flock member is given a fixed set of flock members $\mathbf{S}_i$ (*e.g.*, flock members are nodes in a graph with fixed connectivity).

$\text{DistanceBased}_i(d(\cdot), \mathbf{R}_{[d_{min}, d_{max}]}) = \{j \in \mathbf{D}_i : d_{min} \leq d(i,j) < d_{max}\}$
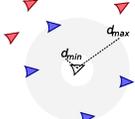
All flock members within some distance from the sensing flock member are selected.

$\text{Nearest}_i(d(\cdot)) = k \in \mathbf{D}_i \text{ where } k \in \text{t}(\underset{j \in \mathbf{D}_i}{\text{argmin}} d(i,j))$
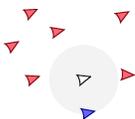
Only the nearest flock member from the sensing flock member is selected.

$\text{VoronoiBased}_i() = \{j \in \mathbf{V}_i \subseteq \mathbf{A} : j\} = \mathbf{V}_i$
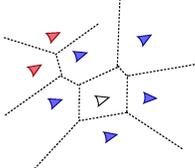
The voronoi neighbors, represented by the set $\mathbf{V}_i(t)$, of the sensing flock member are selected.

$\text{k-Nearest}_i(d(\cdot)) = \{j \in \mathbf{D}_i : \text{k-}\underset{j \in \mathbf{D}_i}{\text{argmin}} d(i,j)\}$
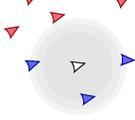
A generalized form of the nearest neighbor perception function which selects the k-nearest flock members from the sensing flock member.

$\text{BoundingBox}_i(\mathbf{u}_1, \mathbf{u}_2) = \{j \in \mathbf{D}_i : (\mathbf{u}_1x \leq \mathbf{r}_jx \leq \mathbf{u}_2x) \wedge (\mathbf{u}_1y \leq \mathbf{r}_jy \leq \mathbf{u}_2y)\}$
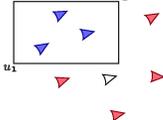
All flock members within a defined bounding box are selected (this function is designed for a 2D plane but it is trivial to extend to higher dimensions).

$\text{Angle}_i(\alpha) = \{j \in \mathbf{D}_i : \arccos(\hat{\mathbf{r}}_j \cdot \hat{\rho}_i) < \alpha\}$
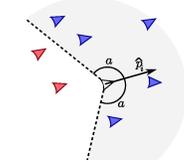
All flock members that are within a specified field of view are selected.

Legend: ◁ Sensed agent ◀ Selected agent ▽ Acting agent

Table 2: Descriptions of perception functions found in the selected literature. We suggest that the majority of neighbor selection stages could be created using a combination of the functions listed here, but these may not be the only possible functions.

or left and right motor speeds for a two wheeled robot). Similar to the sensing stage, there is no formal output given by the physical motion stage; however, the resulting motions have an impact on the visible environment used by the sensing stage (*e.g.* physical motion could affect which flock members belong to $\mathbf{D}_i(t + \Delta t)$). Therefore, we have connected the sensing stage and the physical motion stage in Figure 1.

2.2 The Five Stages: Selected Literature

By applying the DT to the selected literature we can (1) show the five stages are indeed key aspects for the generation of flocking motions, (2) gain a better understanding of the current microscopic flocking motion models, and (3) determine which aspects of flocking motion generation have not been fully studied. This section highlights specific examples found in the literature that have various levels of completeness in regards to specific DT stages. Figure 2 gives examples of specific data-flows seen in the literature between the five stages of the DT.

2.2.1 Sensing

Surveying the literature using the DT shows that different research communities tend to only focus on particular aspects of flocking motion generation. Biology and physics models simplify both the sensing and physical motion stages, where control literature models simplify the sensing stage but introduce motion constraints and various types of noise. There are few works which investigate the sensing stage in great detail, with one example being Kelly and Keating (1996). The treatment of the sensing stage in Kelly and Keating (1996) describes all of the sensors used and the various properties associated with those specific sensors. Additionally, Kelly and Keating (1996) give a detailed description of the physical flock member, which could have an impact on the sensing, flock member detection, and neighbor selection stages. The data-flow between the sensing and flock member detection stages for the model presented by Kelly and Keating (1996) can be seen in Figure 2a.

Although the level of detail given in Kelly and Keating (1996) is desirable, it is not always feasible to describe all five stages in a detailed manner due to space limitations. In contrast to the verbose description in Kelly and Keating (1996), Gazi and Passino (2005) gives a brief, yet complete description of the sensing capabilities of the flock members. Even though the sensing capabilities for the flock members rely on strong assumptions (instantaneous and perfect sensing with an infinite range), the treatment of the sensing stage does not allow for ambiguities in the understanding of the presented approach.

The vast majority of the literature either does not discuss or present a complete description of the sensing stage. An example of a publication that does not treat the sensing stage can be seen in Tanner et al (2003a). Tanner et al (2003a) simply state that there are n flock members moving in a plane that contain a position $(\mathbf{r}_i(t))$ and a velocity $(\mathbf{v}_i(t))$. There is no discussion of how the positions and velocities are sensed and/or calculated. Only from the context could we infer that the model in Tanner et al (2003a) uses an oracle to maintain the information of the flock members.

For the publications that have a partial treatment of the sensing stage (Vicsek et al, 1995; Viscido et al, 2002; Czirók et al, 1997), it is typical to see the following style of description: "The flock member can detect all members within a radius of r". From this description the reader cannot disambiguate between the case where the sensing radius is a simulation of a sensor limitation or if the sensing radius is a part of the motion model. We know from the biology literature that some species only use a limited/specific number of flock members to calculate their next motion, thus the distinction of what the sensing radius actually represents is important for the understanding of the overall model.

2.2.2 Flock Member Detection

The flock member detection stage is one of the least completely treated and discussed aspects of flocking motion generation. The vast majority of the literature does not consider flock member detection and makes the assumption that all of the flock members within the sensing range are included in the set $\mathbf{D}_i(t)$. Additionally, the flock member detection stage is also responsible for the detection of the required information.

If in the sensing stage a *velocity* sensor is not specified and the motion model requires the flock member's velocity, then the flock member detection stage should detail how the velocity information is calculated (*e.g.*, the velocity is inferred from the displacement of $\mathbf{r}_i(t)$ and $\mathbf{r}_i(t + 1)$).

A couple of good treatments of the flock member detection stage can be seen in Turgut et al (2008); Kelly and Keating (1996); Arkin and Balch (1999). Turgut et al (2008) give an adequate description through describing how all of the required information is sensed (*e.g.*, velocity, identification, orientation, etc.) by the flock members. However, Turgut et al (2008) does not discuss if any noise or detection error exist in the flock member detection stage. A common data-flow between the flock member detection and neighbor selection stage can be seen in Figure 2b.

To this point, we have only considered a flock member detection stage that detects real flock members (*i.e.*, $\mathbf{D}_i(t) \subset \mathbf{A}_i(t)$). Another possible role of the flock member detection stage is the detection/creation of virtual flock members. Virtual flock members can be used to avoid obstacles (Olfati-Saber, 2006) and/or to assist in maintaining a desired flock structure (Lindhé et al, 2005). A virtual flock member is created from the sensed information. The information typically used for the creation of virtual flock members is encoded in the set $\mathbf{A}_i(t)$ and any detected obstacles. In our notation, a virtual flock member is a flock member that exist in the set $\mathbf{D}_i(t)$ but does not exist in the set $\mathbf{A}_i(t)$. It is important to note that a virtual flock member in the set $\mathbf{D}_i(t)$ cannot be distinguished from a non-virtual flock member in the same set.

2.2.3   Neighbor Selection

The neighbor selection stage is treated in almost all of the literature and is the only stage that is generally completely described. One of the most complete treatments can be found in Viscido et al (2002). For each of the perception functions, Viscido et al (2002) explicitly states which flock members will be selected and passed to the motion computation stage. The data-flow for one of the models presented by Viscido et al (2002) between the neighbor selection and motion computation stage can be seen in Figure 2c. In addition to which flock members are to be selected, Viscido et al (2002) gives brief explanations for the selection decisions. The one exception to this is in their treatment of the local crowded horizon (LCH) motion model. Viscido et al (2002) do not clarify if all of the detected flock members are selected or if only the members apart of the 'most crowded horizon' are selected.

In contrast to Viscido et al (2002) fairly complete treatment of the neighbor selection stage, Mikhailov and Zanette (1999) do not explicitly treat this stage at all. With that said, it is clear from the context that Mikhailov and Zanette (1999) are using a perception function that selects all of the flock members in the set $\mathbf{D}_i(t)$ (see the perception function All() in Table 2).

The majority of the literature typically uses one perception function in the neighbor selection stage; however, there are some works which use a combination of perception functions. Using the perception functions in Table 2 we see that the motion model proposed by Gueron et al (1996) uses over six perception functions. This is because the motion computation stage for Gueron et al (1996) has a large number of conditions which affect the computation output (similar to the "attraction/repulsion" zone example used earlier). For a more detailed treatment of this work please see Section 4.1.1.

Topological Flocking Models

In recent years, microscopic flocking motion literature has been using the term 'topological' to describe a particular set of flocking motion models (Tanner et al, 2003a,b; Ballerini et al, 2008; Ginelli and Chaté, 2010; Niizato and Gunji, 2011; Bode et al, 2011; Cavagna et al, 2010). Topological motion models only differ from 'metric' motion models in the neighbor selection stage of the DT, however, this distinction is not clear in the majority of the topological literature. Niizato and Gunji (2011) does a good job in teasing out the difference between topological and metric flocking motion models. Additionally, Niizato and Gunji (2011) presents a motion model that utilizes both a metric and topological neighbor selection function.

2.2.4   Motion Computation

Even though the motion computation stage is always treated, the description of this stage is typically incomplete, which causes ambiguities when attempting to formalize or implement a motion model. One

reason for the lack of completeness is from not explicitly addressing all of the outputs generated by the neighbor selection stage. For example, the simple nearest neighbor motion model (Viscido et al, 2002) does not detail the flock members computed motions when there are no neighbors. In other words, if the set $\mathbf{D}_i(t)$ is empty, the motion computation stage is undefined.

Another cause for the lack of incompleteness in the motion computation stage is from the use of vague terminology such as "back-up" and "turn left". As we discuss later in this work (see Section 4.1.3) we see that the use of vague terminology prevents us from formalizing such models, and thus it prevents us from implementing a model as it was intended to be implemented. A third reason for incompleteness of the motion computation stage can be seen in Viscido et al (2002). Within the same publication there are two possible interpretations of the LCH motion model, but the authors never distinguish between the two of them. The first prose description of the LCH motion model and the implemented version of the model are similar but not exactly the same. We have implemented both of these interpretations on a physical system and, fortunately, found that the generated motions only differ slightly (Fine and Shell, 2012). However, the assumptions made by the two different interpretations do have an impact of the complexity of the motion model.

2.2.5 Physical Motion

The physical motion stage is rarely treated in the flocking literature (Ferrante et al, 2012). A common input to this stage can be seen in Figure 2d, where the motion computation stage takes a computed position ($\mathbf{r}_i(t)$) and translates it into low-level motor commands for the flock member.
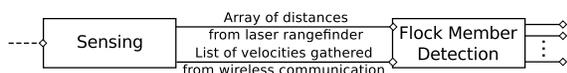
(a) The sensing stage reduces the visible environment into an array of distances from a laser rangefinder and a list of velocities gathered through wireless communication; both being common sensors chosen for robotic applications. The array and list are passed to the flock member detection stage as the only inputs. This particular data-flow is seen in Kelly and Keating (1996).
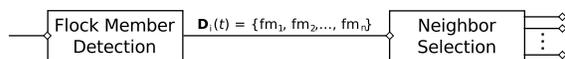
(b) The flock member detection stage translates the information given by the sensing stage into a set of n flock members. It is important to note that $\mathbf{D}_i(t)$ is a set of abstract flock members and that a individual flock member (fm) can be thought of as a feature vector. The features of a flock member are defined by the required attributes (e.g., $\mathbf{r}_j(t)$, $\mathbf{v}_j(t)$, $\theta_j(t)$, etc.). The set of flock members is passed to the neighbor selection stage. This data flow is used in all of the selected literature.
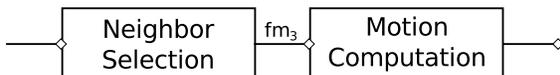
(c) The neighbor selection stage selects a subset ($\mathbf{I}_i(t)$) of the set $\mathbf{D}_i(t)$ to be passed to the motion computation stage. The selection function used in this example only selects the nearest flock member. This example can be seen in Hamilton (1971) and Viscido et al (2002).
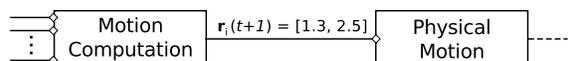
(d) The motion computation stage (i.e., control-law) uses the set $\mathbf{I}_i(t)$ to compute the next motion for the ith flock member. In this example, the next position for the flock member is (1.3, 2.5) and it is passed to the physical motion stage. This example can be seen in Viscido et al (2002), Hamilton (1971), Szabó et al (2008), Szabó et al (2009), and Gazi and Passino (2003).

Figure 2: Specific examples of data-flow between the various stages in the DM. Note that the examples listed here are from a combination of various publications. There is no known example of a publication that unambiguously describes the data-flow between all stages.

3   Current Microscopic Models

In this section we analyze the high-level design choices for the selected microscopic motion models. Table 3 is a categorical view of the sensing and computational requirements of individual flock members as well as the composition of the entire flock (*i.e.*, is the flock heterogeneous or homogeneous). The eight attributes (flock

composition, flock member mobility, continuous/discrete time, collision avoidance, neighbor identification, neighbor's position, neighbor's velocity, and neighbor's orientation) highlighted in Table 3, together, give an adequate description of the model's required information. Additionally, Table 3 allows one to gain a better understanding for if a particular model is biologically feasible, or how easily the model may be to implement on a physical system. For example, if a motion model requires idealize/perfect motion, then the resulting motions of this rule implemented on a physical (*i.e.*, noisy) system may not be equivalent to the desired motions. In general, Table 3 can be used to answer three questions; (1) what is the composition of the flock, (2) how realistic are the constraints, and (3) what sensing information is required?

Paper	Flock Composition	Mobility	Continuous/Discrete	Collision Avoidance	Identification	Position	Velocity	Orientation
Viscido et al (2002)(SNN/HA/LCH)	□	○	⊙	–	–	√	–	–
Conradt et al (2009)	⊞	○	⊖	Member	–	√	√	–
Gueron et al (1996)	⊞	○	⊙	Member	–	√	–	–
Couzin et al (2005)	⊞	○	⊙	Member	–	√	√	–
Lopez et al (2012)	⊞	○	⊙	–	–	√	–	–
Hamilton (1971)	□	○	⊙	–	–	√	–	–
Vicsek et al (1995)	□	○	⊙	–	–	√	–	√
Dong (2012)	□	○	⊙	–	–	√	√	–
Smith and Martin (2009)	□	○	⊙	–	–	√	–	√
Shimoyama et al (1996)	□	○	⊖	Member	–	√	√	√
Czirók et al (1997)	□	○	⊙	–	–	√	–	√
Szabó et al (2008)	□	○	⊙	–	–	√	√	√
Szabó et al (2009)	□	○	⊙	–	–	√	√	√
Levine et al (2000)	□	○	⊖	Member	–	√	–	–
Toner and Tu (1998)	□	○	⊙	–	–	√	–	√
Grégoire et al (2003)	□	○	⊙	Member	–	√	–	√
Camperi et al (2012)	□	○	⊙	Member	–	√	√	–
Helbing et al (2005)	⊞	○	⊖	All	–	√	√	–
Matarić (1993)	□	△	⊖	All	–	√	–	–
Reynolds (1987)	□	△	⊖	All	–	√	√	√
Kelly and Keating (1996)	□	△	⊖	All	√	√	√	√
Turgut et al (2008)	□	△	⊙	All	–	√	√	–
Gökçe and Şahin (2009)	⊞	△	⊙	All	–	√	√	√
Tanner et al (2003a)	□	○	⊖	Member	√[‡]	√	√	√
Tanner et al (2003b)	□	○	⊖	Member	–	√	√	√
Jadbabaie et al (2002)	□	○	⊙	–	–	√	–	√
Gazi and Passino (2005)	□	○	⊙	–	√	√	–	–
Gazi and Passino (2003)	□	○	⊙	Member	–	√	√	√
Olfati-Saber (2006)	□	○	⊙	All	–	√	–	√
Arkin and Balch (1999)	□	△	⊖	All	√	√	–	–
Fine and Shell (2011)	□	△	⊙	–	–	√	–	–
Hauert et al (2011)	□	△	⊙	Member	–	√	√	–

Legend: □ Homogeneous ⊞ Heterogeneous ○ Idealized △ Constrained ⊖ Continuous ⊙ Discrete

Table 3: A categorical review of the information and flock member requirements for each of the selected motion models. This table can be used as a first pass to identify which models could be useful when designing a new model or when identifying models that may benefit from additional investigations. Entries marked with a (√) signifies that the particular attribute is utilized. For the collision avoidance attribute, (Member) signifies that the model only considers member to member collision avoidance, while (All) signifies that the model considers both member and environment collision avoidance.

[‡]Identification is only used in the perception function.

3.1 Definitions of Table 3 Attributes

3.1.1 Flock Composition

Several flocking motion studies investigate the possibility of differences (sometimes subtle, sometime significant) in the flock members. Apart from physical differences (*e.g.*, in sensing capability, size, appearance), there can be differences in the underlying control law; more precisely, the motion computation stage in the DT may vary between flock members. A flock's composition can either be homogeneous ($\square$), where all flock members have identical motion computation stages, or heterogeneous ($\boxplus$), where at least one flock member has a different motion computation stage.

   This definition of group composition only considers the motion computation stage when determining if a flock is homogeneous or not, and does not consider the motions exhibited by individual flock members. For example, if all flock members have an identical probabilistic motion model (Viscido et al, 2002; Shimoyama et al, 1996), then the flock is considered homogeneous even though the motions of the individual flock members given identical input may not be the same. However, if the model has parameters which are unique to a subset of flock members (Conradt et al, 2009; Couzin et al, 2005), we consider the flock to be heterogeneous because the unique parameters can drastically affect the flock member's motion. In the case of Couzin et al (2005) there are two types of flock members (informed and uninformed), thus $\mathbf{A}_i(t)$ can be partitioned into two distinct subsets of flock members, and therefore considered to be heterogeneous.

   For this publication, we do not consider either the sensor configuration or the physical appearance of the flock members in regards to the flock's composition, since we are only interested in the chosen motion model. However, a variation of flock composition could consider the sensor configuration of the flock members. Knowing the particular sensor configuration may reveal implementation details that may affect the design of the model. Even though we do not consider this here, the sensor configuration should be discussed when presenting the sensing stage. Studies have also investigated the effects of heterogeneous flock members (in appearance) on the human perception of a flock (Ip et al, 2006). We do not consider heterogeneous flocks (in appearance) because we assume that the physical appearance of the flock members (for the selected literature) does not affect the exhibited flocking motions.

3.1.2 Mobility

The mobility of the flock details the physical motion individual flock members can perform and we assume that all of the flock members have the same mobility (*i.e.*, the flock is homogeneous in regards to mobility). A flock member's mobility can either be idealized ($\bigcirc$), where the flock member can immediately and perfectly perform the computed motion, or constrained ($\triangle$), where the flock member has a restricted set of possible motions (*e.g.*, grid-based motion) or imperfect (noisy) motion. For example, if a flock member that has idealized motion executes a model that calculates the next position ($\mathbf{r}_i(t+1)$) at time t, we know that the flock member will arrive at $\mathbf{r}_i(t+1)$ (with no error). Conversely, if that same flock member now has constrained motion, there could be a difference between the calculated and observed $\mathbf{r}_i(t+1)$ (*i.e.*, the flock member is not guaranteed to arrive at $\mathbf{r}_i(t+1)$).

3.1.3 Discrete/Continuous

Flock members can execute the given model in either discrete time ($\odot$) or in continuous time ($\ominus$). Flock members operating in discrete time only sense, compute, and act (move) in distinct time intervals (*i.e.*, a flock member may only sense, compute, and act every 0.1 seconds, regardless of the sensors sensing frequency). Conversely, flock members that execute the given model in continuous time will sense, compute, and act at the frequency of the given sensors and the time it takes to preform the computations. The important distinction between the two time choices is that in discrete time, the flock members may ignore information if the time interval is too large, where in continuous time, no information will be ignored (assuming all sensors have the same frequency). However, if the information from the sensors is noisy and the flock member is operating in continuous time, the resulting motion may be sporadic and may not exhibit the desired motions.

3.1.4 Collision Avoidance

The collision avoidance of the flock details the individual flock member's ability to avoid various types of collisions. The two types of collisions studied in the literature are member-member collisions (a flock member collides with another flock member) and member-environment collisions (a flock member collides with an object in the environment that is not another flock member). Flock members can avoid collisions with other flock members (Member), the environment (Environmental), both flock member and environmental obstacles (All), or flock members have no collision avoidance capabilities (–).

3.1.5 Identification

Motion models that utilize identification assume that each flock member has a unique label (ID) (*e.g.*, 'leader' or 'flock member 42'). Furthermore, these models assume that each flock member has the ability to 'detect' other flock member's ID at any time. Identification can be used to assist in member-to-member communication as well as allowing 'follower' flock members to identify the 'leader' flock member(s).

It is important to note, if a flock member has the ability to identify other members, this does not imply that the member has the ability to associate information between sensing steps (for the definition of association see Section 6). Furthermore, we must note that the index of the flock members in the sets $\mathbf{A}_i(t)$ and $\mathbf{D}_i(t)$ cannot be used for identification (*i.e.*, the flock member $j \in \mathbf{D}_i(t)$ is not guaranteed to be member $j \in \mathbf{D}_i(t+1)$). Therefore, if identification is required, there must be an identification attribute associated with each member, just as all other attributes (*e.g.*, position, velocity).

3.1.6 Position, Velocity, and Orientation

The position, velocity, and orientation columns identify whether or not a model utilizes that particular type of information. These three columns do not distinguish how the information is sensed or computed (*e.g.*, global versus local reference frame), rather they simple state what information is required. In other words, these columns identify the minimum informational requirements the motion model must have in order to compute the next motion for the flock member.

3.2 Observations from Table 3

3.2.1 Group Composition

Observing Table 3, we see that six of the investigations consider heterogeneous flock members. Even though we have a limited definition of heterogeneity, all of the studies which consider homogeneous flock members, use truly homogeneous members (*i.e.*, all aspects of the flock members are identical). For the publications which do consider heterogeneous flocks, the studies typically investigate how *informed members* or *leaders* can affect the motions of the flock (Couzin et al, 2005; Conradt et al, 2009). These investigations have parameters which differ according to the flock member's classification (*e.g.*, leader or follower). In Gueron et al (1996), heterogeneity represents *strong* and *weak* flock members, where *strong* flock members move faster then *weak* members. We do not consider this to be heterogeneity with respect to mobility because the feasible motions of the flock members are still the same, one type of flock member simply performs the motion faster than the other (i.e., a different gain in the motion computation stage).

3.2.2 Mobility

We observed eight publications which consider constrained motion, which all exist in the robotics and control literature. The majority of the literature assumes that if a flock member computes it's next position as $\mathbf{r}$, then at time $t + 1$ the flock member will be at $\mathbf{r}$ (with no error). Of the literature that does consider constrained motion, none of the models explicitly handle the motion constraints in any of the DT stages. We must note that we chose to classify Vicsek et al (1995) as using idealized motion. Even though this model is probabilistic in regards to the motion noise, the noise is added in the motion computation stage.

When the flock member moves to the calculated position in the physical motion stage, there is no error; thus, the Vicsek et al (1995) model is consider to use idealized motion.

3.2.3   Continuous/Discrete Time

Table 3 shows a divide among research groups with respect to the use of continuous or discrete time models. Almost all of the selected literature from the biology and physics communities use discrete time, where the robotics and control groups predominately use continuous time. One possible cause for this dichotomy are the chosen forms of validation; see Table 6 (*e.g.*, computer simulation versus physical implementation).

Continuous time models are typically more complete (in the specific sense outlined in the Introduction) than discrete time models, although, there are some exceptions. Gueron et al (1996) is an example of a discrete time motion model that is completely described. As we see in Gueron et al (1996), the authors took great care to present what motion output would occur given any possible input. Another example of a completely described discrete time model can be found in Conradt et al (2009), where the authors detail what flock members do in the absence of neighbors (*i.e.*, the neighbor selection stage outputs an empty list).

3.2.4   Collision Avoidance

Eight of the publications consider both environmental and flock member collision avoidance, with nine other publications considering member-member collision avoidance. The flock members in the Olfati-Saber (2006) study perform collision avoidance with both the environment and other flock members; however, the motion computation stage does not handle any environmental obstacles. Flock members generate virtual flock members (see Section 6) which travel along the boundary of the detected obstacles, thus the motion computation stage only considers flock member avoidance. Additionally, none of the investigations explicitly describe, with the exception of Reynolds (1987), how the obstacle avoidance is performed.

3.2.5   Position/Velocity/Orientation

When considering the complexity of a given motion model, we only consider if the model requires position, velocity, and/or orientation information. There does not appear to be a difference in the design of the motion computation stage with respect to the way in which the information in sensed (*e.g.*, global, local, or inferred). However, there is a difference in how the sensed information is used.

The majority of the motion models use the information in the motion computation stage, but some models (Vicsek et al, 1995; Jadbabaie et al, 2002) use some of the information in the neighbor selection stage. In Viscido et al (2002) the motion computation stage only requires the velocities ($\mathbf{v}_j(t)$) of the selected flock members; however, position information ($\mathbf{r}_i(t)$) is used to select a subset of flock members from $\mathbf{A}_i(t)$. In Table 3 we do not make a distinction on where the information is used.

With exception of the prose description of the LCH motion rule in Hamilton (1971), all of the selected models require position information. In the proposed, but not validated, description of the LCH motion rule each flock member moves towards the center of the highest density of detected flock members. With respect to velocity and orientation information, there does not seem to be any major trends seen in the literature.

4   Specification of Motion Rules

To aid in the comparison of various flocking motion models and to assist in the understanding of the current state of the literature, we translated the selected models into a common notation and formalized stages 3 and 4, which together create the motion rule. Table 4 shows the formalization of the neighbor selection stage and motion computation stage for each of the selected models. Observing Table 4 reveals that even though each of these models have the same aim, there are many ways in which the flocking problem can be solved.

Paper	Neighbor Selection	Motion Computation
Viscido et al (2002) (SNN)	$k = \text{Nearest}_i(L^2)$	$\mathbf{r}_i(t + \Delta t) = \begin{cases} \mathbf{r}_i(t) + \hat{\mathbf{d}}_{ik}(t) & \text{if } L^2(\mathbf{r}_i(t), \mathbf{r}_k(t)) > r_\rho, \\ \mathbf{r}_i(t) & \text{otherwise.} \end{cases}$
Viscido et al (2002) (HA)	$k = \text{Nearest}_i(L^2)$ $l = \text{Nearest}_k(L^2)$	$\mathbf{r}_i(t + \Delta t) = \begin{cases} \mathbf{r}_i(t) + \hat{\mathbf{e}}(\mathbf{t}) & \text{if } L^2(\mathbf{r}_i(t), \mathbf{r}_k(t)) > r_\rho, \\ \perp (\mathbf{r}_i(t) + \hat{\mathbf{e}}(\mathbf{t})) & \text{otherwise.} \end{cases}$ $\mathbf{e}(t) = (\hat{\mathbf{d}}_{lk}(t) + \mathbf{r}_l(t)) - \mathbf{r}_i(t)$
Viscido et al (2002) (LCH)[§] $\mathbf{I}_i = \text{All}()$		$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \hat{\mathbf{e}}$ $\mathbf{e} = \sum_{j \in \mathbf{I}_i} \dfrac{\mathbf{d}_{ij}}{1 + w_1 L^2(\mathbf{r}_i(t),\ \mathbf{r}_j(t))}$ Where w_1 has units of $\frac{1}{\text{distance}}$
Conradt et al (2009)	$\mathbf{I}_{i\alpha} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0,\ \alpha]})$ $\mathbf{I}_{i\beta} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[\alpha,\ \beta]})$	$\mathbf{d}_i(t) = \begin{cases} -\dfrac{1}{|\mathbf{I}_{i\alpha}|} \sum\limits_{j \in \mathbf{I}_{i a}} \dfrac{\mathbf{r}_j(t) - \mathbf{r}_i(t)}{L^2(\mathbf{r}_i(t), \mathbf{r}_j(t))} & \text{if } |\mathbf{I}_{i\alpha}| \neq 0, \\ = \dfrac{1}{4|\mathbf{I}_{i\beta}|} \sum\limits_{j \in \mathbf{I}_{i\beta}} \dfrac{\mathbf{r}_j(t) - \mathbf{r}_i(t)}{L^2(\mathbf{r}_j(t), \mathbf{r}_i(t))} + \dfrac{1}{4|\mathbf{I}_{i\beta}|} \sum\limits_{j \in \mathbf{I}_{i\beta}} \hat{\mathbf{v}}_j(t) + w_1^i \dfrac{\mathbf{r}_*(t) - \mathbf{r}_i(t)}{2 L^2(\mathbf{r}_*(t), \mathbf{r}_i(t))} & \text{if } |\mathbf{I}_{i\beta}| \neq 0, \\ \dfrac{\mathbf{r}_*(t) - \mathbf{r}_i(t)}{|\mathbf{r}_*(t) - \mathbf{r}_i(t)|} & \text{otherwise.} \end{cases}$
Gueron et al (1996)	$\mathbf{I}_S = \text{BoundingBox}_i((\mathbf{r}_i x - w_1, \mathbf{r}_i x + w_1), (\mathbf{r}_i y - w_2, \mathbf{r}_i y + w_2))$ $\mathbf{F} = \text{Angle}_i() \cap \mathbf{I}_S$ $\mathbf{LF} = \text{Angle}_i() \cap \mathbf{I}_S$ $\mathbf{RF} = \text{Angle}_i() \cap \mathbf{I}_S$ $\mathbf{LB} = \text{Angle}_i() \cap \mathbf{I}_S$ $\mathbf{RB} = \text{Angle}_i() \cap \mathbf{I}_S$	$\theta_i(t + \Delta t) = \begin{cases} \theta_i(t) & \text{if } (\mathbf{F} \wedge \neg\mathbf{LF} \wedge \neg\mathbf{RF} \wedge \neg\mathbf{LB} \wedge \neg\mathbf{RB}) \vee \\ & (\neg\mathbf{F} \wedge \mathbf{LF} \wedge \mathbf{RF} \wedge \neg\mathbf{LB} \wedge \neg\mathbf{RB}) \vee \\ & (\neg\mathbf{F} \wedge \neg\mathbf{LF} \wedge \neg\mathbf{RF} \wedge \mathbf{LB} \wedge \mathbf{RB}), \\ \theta_i(t) - 90° & \text{if } (\neg\mathbf{F} \wedge \mathbf{LF} \wedge \neg\mathbf{RF} \wedge \neg\mathbf{LB} \wedge \neg\mathbf{RB}) \vee \\ & (\neg\mathbf{F} \wedge \neg\mathbf{LF} \wedge \neg\mathbf{RF} \wedge \mathbf{LB} \wedge \neg\mathbf{RB}), \\ \theta_i(t) + 90° & \text{if } (\neg\mathbf{F} \wedge \neg\mathbf{LF} \wedge \mathbf{RF} \wedge \neg\mathbf{LB} \wedge \neg\mathbf{RB}) \vee \\ & (\neg\mathbf{F} \wedge \neg\mathbf{LF} \wedge \neg\mathbf{RF} \wedge \neg\mathbf{LB} \wedge \mathbf{RB}). \end{cases}$ $u_i(t + \Delta t) = \begin{cases} u_i(t)\frac{1}{w_i} & \text{if } (\mathbf{F} \wedge \neg\mathbf{LF} \wedge \neg\mathbf{RF} \wedge \neg\mathbf{LB} \wedge \neg\mathbf{RB}), \\ u_i(t) & \text{if } (\neg\mathbf{F} \wedge \mathbf{LF} \wedge \neg\mathbf{RF} \wedge \neg\mathbf{LB} \wedge \neg\mathbf{RB}) \vee \\ & (\neg\mathbf{F} \wedge \neg\mathbf{LF} \wedge \neg\mathbf{RF} \wedge \mathbf{LB} \wedge \neg\mathbf{RB}) \vee \\ & (\neg\mathbf{F} \wedge \neg\mathbf{LF} \wedge \mathbf{RF} \wedge \neg\mathbf{LB} \wedge \neg\mathbf{RB}) \vee \\ & (\neg\mathbf{F} \wedge \neg\mathbf{LF} \wedge \neg\mathbf{RF} \wedge \neg\mathbf{LB} \wedge \mathbf{RB}), \\ 0 & \text{if } (\neg\mathbf{F} \wedge \mathbf{LF} \wedge \mathbf{RF} \wedge \neg\mathbf{LB} \wedge \neg\mathbf{RB}), \\ u_i(t)w_i & \text{if } (\neg\mathbf{F} \wedge \neg\mathbf{LF} \wedge \neg\mathbf{RF} \wedge \mathbf{LB} \wedge \mathbf{RB}). \end{cases}$ See Section 4.1.1

See Section 4.1.1

[§]This is the formalization of the LCH motion rule that was validated in Viscido et al (2002). The other version of the motion rule is discussed in more detail through out this meta-study (see Section 3.2.5).

Paper	Neighbor Selection	Motion Computation
Couzin et al (2005)	$\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0,\beta]})$	$\mathbf{d}_i(t + \Delta t) = \frac{\hat{\mathbf{e}} + w_1 \mathbf{d}_{i*}(t)}{|\hat{\mathbf{e}} + w_1 \mathbf{d}_{i*}(t)|}$ $\mathbf{e} = \sum_{j \in \mathbf{I}} \frac{\mathbf{r}_j(t) - \mathbf{r}_i(t)}{|\mathbf{r}_j(t) - \mathbf{r}_i(t)|} + \sum_{j \in \mathbf{I}} \frac{\mathbf{v}_j(t)}{|\mathbf{v}_j(t)|}$
Lopez et al (2012)	$\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0,\beta]})$	$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + w_1 \Delta t * \theta_i(t + \Delta t)$ $\theta_i(t+\Delta t) = \frac{1}{|\mathbf{I}|} \sum_{j \in \mathbf{I}} w_2(|r_j(t) - r_i(t)|)\theta_j(t) + \frac{1}{|\mathbf{I}|} \sum_{j \in \mathbf{I}} w_3(|\mathbf{r}_j(t) - \mathbf{r}_i(t)|) \frac{\mathbf{r}_j(t) - \mathbf{r}_j(t)}{|\mathbf{r}_j(t) - \mathbf{r}_j(t)|} + \eta_i(t)$ Where $\eta_i(t) \doteq$ a stochastic component
Hamilton (1971) (1D)	$\mathbf{I} = \text{VoronoiBased}_i()$ $k = \text{VoronoiBased}_{\mathbf{I}_1}()\backslash i$ $l = \text{VoronoiBased}_{\mathbf{I}_2}()\backslash i$	$\mathbf{r}_i(t+\Delta t)\begin{cases} \mathbf{r}_i(t) & \text{if } L^2(\mathbf{r}_{\mathbf{I}_0}(t), \mathbf{r}_{\mathbf{I}_1}(t)) < min(L^2(\mathbf{r}_l(t), \mathbf{r}_i(t)), L^2(\mathbf{r}_k(t), \mathbf{r}_i(t))), \\ \frac{\mathbf{r}_l(t) - \mathbf{r}_i(t)}{2} & \text{if } L^2(\mathbf{r}_l(t), \mathbf{r}_i(t)) < L^2(\mathbf{r}_k(t), \mathbf{r}_i(t)), \\ \frac{\mathbf{r}_k(t) - \mathbf{r}_i(t)}{2} & \text{otherwise.} \end{cases}$
Hamilton (1971) (2D)	$k = \text{Nearest}_i(L^2, \mathbf{R}_{[0,\ \beta]})$	$\mathbf{r}_i(t+\Delta t) = \begin{cases} \mathbf{r}_i(t) + \hat{\mathbf{d}}_{ik}(t) & \text{if } L^2(\mathbf{r}_i(t), \mathbf{r}_k(t)) > r_\rho, \\ \mathbf{r}_i(t) & \text{otherwise.} \end{cases}$
Vicsek et al (1995); Czirók et al (1997); Smith and Martin (2009)¶	$\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0,\ \beta]})$	$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t$ $\theta_i(t + \Delta t) = \sum_{j \in \mathbf{I}} \arctan\left(\frac{\sin\theta_j(t)}{\cos\theta_j(t)}\right) + \Delta\theta$ Where $\Delta\theta \in \mathcal{U}(-\frac{\eta}{2},\ \frac{\eta}{2})$
Dong (2012)	$\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0,\ \beta]})$	$\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t$ $\mathbf{v}_i(t + \Delta t) = \sum_{j \in \mathbf{I}} \Delta t a_{ij}(L^2(\mathbf{r}_i(t), \mathbf{r}_j(t)))(\mathbf{v}_j(t) - \mathbf{v}_i(t)) + \mathbf{v}_i(t)$ $a_{ij}(x) = \begin{cases} 1 & \text{if } x \le w_1, \\ 0 & \text{otherwise.} \end{cases}$ Where $w_1 > 0$
Shimoyama et al (1996)	$\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0,\ \beta]})$	$\dot{\mathbf{v}}_i(t) = \frac{1}{m}(-w_1 \mathbf{v}_i(t) + w_2 \mathbf{n}_i(t) + \sum_{j \in \mathbf{I}} \alpha_{ij}(t)\mathbf{f}_{ij}(t) + \mathbf{g}_i(t))$

¶Uses a common absolute velocity.

Paper	Neighbor Selection	Motion Computation
(Also based on Sugawara (2012))		$\dot{\mathbf{n}}_i(t) = \frac{1}{\tau}(\mathbf{n}_i(t) \times \hat{\mathbf{v}}_i(t) \times \mathbf{n}_i(t))$ $\alpha_{ij}(t) = 1 + d\left[\mathbf{n}_i(t) \cdot \frac{\mathbf{r}_j(t) - \mathbf{r}_i(t)}{\|\mathbf{r}_j(t) - \mathbf{r}_i(t)\|}\right]$ $\mathbf{f}_{ij}(t) = -w_3\left[\left(\frac{\|\mathbf{r}_j(t) - \mathbf{r}_i(t)\|}{w_4}\right)^{-3} - \left(\frac{\|\mathbf{r}_j(t) - \mathbf{r}_i(t)\|}{w_4}\right)^{-2}\right] \times \left(\frac{\mathbf{r}_j(t) - \mathbf{r}_i(t)}{w_4}\right) \exp\left(\frac{-\|\mathbf{r}_j(t) - \mathbf{r}_i(t)\|}{w_4}\right)$ $\mathbf{g}_i(t) = w_5\left(\frac{\mathbf{e}_i(t) - \mathbf{r}_i(t)}{\|\mathbf{I}\|\|\mathbf{e}_i(t) - \mathbf{r}_i(t)\|}\right)$ $\mathbf{e}_i(t) = \frac{\sum_{j \in \mathbf{I}} \mathbf{r}_j(t)}{\|\mathbf{I}\|}$ Where $m \doteq$ agent's mass, $\tau \doteq$ rotational relaxation time, and $(0 \leq d \leq 1)$ for $\alpha_{ij}(t)$.
Szabó et al (2008, 2009)	$\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0,\ \beta]})$	$\mathbf{e}_i(t + \Delta t) = \nu \cdot \mathcal{M}(\gamma, \xi) \cdot N(s \cdot \frac{\sum_{j \in \mathbf{I}} \mathbf{v}_j(t)}{\|\mathbf{I}\|} + (1 - s) \cdot a(t)\Delta t)$ $\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{e}_i(t)\Delta t$ Where $\nu \doteq \|\mathbf{v}\|$, $\mathcal{M}(e, \xi) \doteq$ rational tensor representing random perturbation with $\gamma :=$ random unit vector chosen uniformly vectors $\perp N(\frac{\sum_{j \in \mathbf{I}} v_j(t)}{\|I\|})$ and $\xi \in \mathcal{U}(-\eta\pi, \eta\pi), N(e) = \frac{e}{\|e\|}, s \in (0, 1], a(t) = \frac{\mathbf{v}(t) - \mathbf{v}(t - \Delta t)}{\Delta t}$
Levine et al (2000)	$\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0,\ \beta]})$	$\hat{\mathbf{v}}_i(t) = \frac{1}{m}\left(w_1\hat{\mathbf{f}}_i(t) - w_2\mathbf{v}_i(t) - \nabla U_i(t)\right)$ $\hat{\mathbf{f}}_i(t) = \sum_{j \in \mathbf{I}} \hat{\mathbf{v}}_j(t) \exp\left(-\frac{\|\mathbf{r}_i(t) - \mathbf{r}_j(t)\|}{w_3}\right)$ $U_i(t) = \sum_{j \in \mathbf{I}} w_4 \exp\left(-\frac{\|\mathbf{r}_i(t) - \mathbf{r}_j(t)\|}{w_5}\right) - \sum_{j \in \mathbf{I}} w_6 \exp\left(-\frac{\|\mathbf{r}_i(t) - \mathbf{r}_j(t)\|}{w_7}\right)$
Toner and Tu (1998)	$\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0,\ \beta]})$	$\theta_i(t + \Delta t) = \sum_{j \in \mathbf{I}} \theta_j(t) + \eta_i(t)$ $\mathbf{r}_i(t + \Delta t) = \mathbf{r}_i(t) + \mathbf{e}$ where $\mathbf{e} = [\cos\theta_i(t + \Delta t), \sin\theta_i(t + \Delta t)]$
Grégoire et al (2003)	$\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0,\ \beta]})$	**Model 1:** $\theta_i(t + \Delta t) = \arg\left[w_1 \sum_{j \in \mathbf{I}} \mathbf{v}_j(t) + w_2 \sum_{j \in \mathbf{I}} \mathbf{f}_{ij}\right] + \eta\xi_i(t)$ **Model 2:** $\theta_i(t + \Delta t) = \arg\left[w_1 \sum_{j \in \mathbf{I}} \mathbf{v}_j(t) + w_2 \sum_{j \in \mathbf{I}} \mathbf{f}_{ij} + \|\mathbf{I}\|\eta\mathbf{e}_i(t)\right]$ $\mathbf{f}_{ij} = \frac{\mathbf{r}_j(t) - \mathbf{r}_i(t)}{L^2(\mathbf{r}_j(t) - \mathbf{r}_i(t))}\begin{cases} -\infty & \text{if } L^2(\mathbf{r}_j(t) - \mathbf{r}_i(t)) < w_3), \\ \frac{1}{4}\frac{L^2(\mathbf{r}_j(t) - \mathbf{R}_i(t)) - w_4}{w_5 - w_4} & \text{if } w_3 < L^2(\mathbf{r}_j(t) - \mathbf{r}_i(t)) < w_5, \\ 1 & \text{otherwise.} \end{cases}$

Paper	Neighbor Selection	Motion Computation
Camperi et al (2012)	$\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0, \beta]})$	$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_0\hat{\mathbf{e}}$ $\mathbf{e} = w_1 \sum\limits_{j \in \mathbf{I}} \mathbf{v}_j(t) + w_2 \sum\limits_{j \in \mathbf{I}} \mathbf{f}_{ij} + \|\mathbf{I}\| \eta \mathbf{e}_i(t)$ $\mathbf{f}_{ij} = \dfrac{\mathbf{r}_j(t) - \mathbf{r}_i(t)}{L^2(\mathbf{r}_j(t) - \mathbf{r}_i(t))} \begin{cases} -\infty & \text{if } L^2(\mathbf{r}_j(t) - \mathbf{r}_i(t)) < w_3, \\ \frac{1}{4}\frac{L^2(\mathbf{r}_j(t) - \mathbf{R}_i(t)) - w_4}{w_5 - w_4} & \text{if } w_3 < L^2(\mathbf{r}_j(t) - \mathbf{r}_i(t)) < w_5, \\ 1 & \text{otherwise.} \end{cases}$ $\mathbf{v}_0 \doteq \text{constant speed of member}$
Helbing et al (2005)		See Section 4.1.2
Matarić (1993)		See Section 4.1.3
Reynolds (1987)		See Section 4.1.4
Kelly and Keating (1996)		See Section 4.1.5
Turgut et al (2008)	$\mathbf{I} = \text{DistanceBased}(L^2, \mathbf{R}_{[0, \beta]})$	$\mathbf{d}_i(t + \Delta t) = \hat{\mathbf{e}}$ $\mathbf{e} = \sum\limits_{j \in \mathbf{I}(t)} e^{\tilde{\theta}_j(t)} + w_1\frac{1}{8}\sum\limits_{k=1}^{8} f_k e^{\frac{4}{\pi}k}$ $f_k = \begin{cases} -\frac{(L^2(\mathbf{r}_i(t) - \mathbf{r}_{O_k}(t)) - r_\rho)^2}{w_2} & \text{if } L^2(\mathbf{r}_i(t) - \mathbf{r}_{O_k}(t)) \geq r_\rho, \\ \frac{(L^2(\mathbf{r}_i(t) - \mathbf{r}_{O_k}(t)) - r_\rho)^2}{w_2} & \text{otherwise.} \end{cases}$ Where $\tilde{\theta}_j(t) = \angle\left(e^{\left(\theta_i(t) - \theta_j(t) + \frac{\pi}{2}\right)} + w_3 e^\eta\right)$, and $\eta = \mathcal{N}\left(\sum\limits_{j \in \mathbf{I}_i(t)}\left(\theta_i(t) - \theta_j(t) + \frac{\pi}{2}\right), \sigma\right)$
Gökçe and Şahin (2009)	$\mathbf{I} = \text{DistanceBased}(L^2, \mathbf{R}_{[0, \beta]})$	$\mathbf{d}_i(t + \Delta t) = \hat{\mathbf{e}}$ $\mathbf{e} = \sum\limits_{j \in \mathbf{I}(t)} e^{\tilde{\theta}_j(t)} + w_1\frac{1}{8}\sum\limits_{k=1}^{8} f_k e^{\frac{4}{\pi}k} + w_2(\mathbf{d}_{i*}(t) - \mathbf{d}_i(t))$ $f_k = \begin{cases} -\frac{(L^2(\mathbf{r}_i(t) - \mathbf{r}_{O_k}(t)) - r_\rho)^2}{w_3} & \text{if } L^2(\mathbf{r}_i(t) - \mathbf{r}_{O_k}(t)) \geq r_\rho, \\ \frac{(L^2(\mathbf{r}_i(t) - \mathbf{r}_{O_k}(t)) - r_\rho)^2}{w_3} & \text{otherwise.} \end{cases}$ Where $\tilde{\theta}_j(t) = \theta_i(t) - \theta_j(t) + \frac{\pi}{2}$ and $\mathbf{r}_{O_k}(t) \doteq$ the pose of the kth obstacle.
Tanner et al (2003a)	$\mathbf{I} = \text{FixedSet}_i()$	$\dot{\mathbf{r}}_\mathbf{i} = \mathbf{V}_i(t)$

Paper	Neighbor Selection	Motion Computation
		$\dot{\mathbf{V}}_{\mathbf{i}} = \mathbf{u}_i(t)$
		$\mathbf{u}_i(t) = \sum\limits_{j \in \mathbf{I}} (\mathbf{v}_i(t) - \mathbf{v}_j(t)) - \sum\limits_{j \in \mathbf{I}} \nabla_{\mathbf{r}_i(t)} U_{ij}$ $U_{ij} = \begin{cases} i\frac{1}{(L^2)^2 + \log(L^2)^2} & L^2 < r_\rho \\ w_1 & L^2 \geq r_\rho \end{cases}$
Tanner et al (2003b)		$\dot{\mathbf{r}}_{\mathbf{i}} = \mathbf{V}_i(t)$
	$\mathbf{I} = \text{DistanceBased}(L^2, \mathbf{R}_{[0, \beta]})$	$\dot{\mathbf{V}}_{\mathbf{i}} = \mathbf{u}_i(t)$ $\mathbf{u}_i(t) = \sum\limits_{j \in \mathbf{I}} (\mathbf{v}_i(t) - \mathbf{v}_j(t)) - \sum\limits_{j \in \mathbf{I}} \nabla_{\mathbf{r}_i(t)} U_{ij}$ $U_{ij} = \begin{cases} i\frac{1}{(L^2)^2 + \log(L^2)^2} & L^2 < r_\rho \\ w_1 & L^2 \geq r_\rho \end{cases}$
Jadbabaie et al (2002) (Leaderless)	$\mathbf{I} = \text{DistanceBased}(L^2, \mathbf{R}_{[0, \beta]})$	$\theta_i(t + \Delta t) = \frac{1}{w_1}(|\mathbf{I}|\theta_i(t) + \sum\limits_{j \in \mathbf{I}} \theta_j(t))$ Where $w_1 > |\mathbf{I}|$
Jadbabaie et al (2002) (Leader-Follower)	$\mathbf{I}_f = \text{DistanceBased}(L^2, \mathbf{R}_{[0, \beta]})$ $\mathbf{I}_l = \text{FixedSet}()$	$\theta_i(t + \Delta t) = \frac{1}{1 + |\mathbf{I}_f \cap \mathbf{I}_l|}(\theta_i(t) + \sum\limits_{j \in \mathbf{I}_f \cap \mathbf{I}_l} \theta_j(t))$
Gazi and Passino (2003)	$\mathbf{I} = \text{All}()$	$\mathbf{r}_i(t + \Delta t) = \sum\limits_{j \in \mathbf{I}} f(L^2(\mathbf{r}_i(t), \mathbf{r}_j(t))$ where $f(x) = -x(w_1 - w_2 exp(\frac{\sqrt{x^\top x}^2}{w_3}))$ with $w_2 > w_1$
Gazi and Passino (2005)	$\mathbf{I} = \text{VoronoiBased}_i()$ $k = \text{VoronoiBased}_{\mathbf{I}_1}()\backslash i$ $l = \text{VoronoiBased}_{\mathbf{I}_2}()\backslash i$	$x_i(t + \Delta t) = \begin{cases} x_i(0) & \text{if } i = 1, \\ \max\left(\begin{array}{l} x_{i-1}(t) + w_1, \\ \min\left(\begin{array}{l} x_i(t) - g\left(x_i(t) - \frac{x_{i-1}(\tau_{i-1}) + x_{i+1}(\tau_{i+1})}{2}\right), \\ x_{i-1}(t) - w_1 \end{array}\right) \end{array}\right) & \text{if } i \neq 1, N, \\ \max\left(\begin{array}{l} x_{N-1}(t) + w_1, \\ x_N(t) - g(x_N(t) - x_{N-1}(\tau_{N-1}) - r) \end{array}\right) & \text{if } i = N. \end{cases}$ Where $g(\cdot) \doteq$ a scaling function and $\tau_i \doteq$ the time when x_i was last sensed.
Olfati-Saber (2006)	$\mathbf{I} = \text{DistanceBased}(L^2, \mathbf{R}_{[0, \beta]})$	$\mathbf{v}_i(t + \Delta t) = \sum\limits_{j \in \mathbf{I}} \phi_\alpha(||\mathbf{r}_j(t) - \mathbf{r}_i(t)||_\sigma)\mathbf{n}_{ij} + \sum\limits_{j \in \mathbf{I}} a_{ij}(\mathbf{v}_j(t) - \mathbf{v}_i(t)) + f_i^\gamma$

18

Paper	Neighbor Selection	Motion Computation

$$\phi_\alpha(x) = ph\left(\frac{x}{||r||_\sigma}\right)\frac{1}{2}\left[(w_1 + w_2)\left(\frac{x + w_3}{\sqrt{1 + (x + w_3)^2}}\right) + (w_1 - w_2)\right]$$

$$||x||_\sigma = \frac{1}{\epsilon}\left[\sqrt{1 + \epsilon||x||^2} - 1\right]$$

$$\mathbf{n}_{ij} = \frac{\mathbf{r}_j(t) - \mathbf{r}_i(t)}{\sqrt{1 + \epsilon||\mathbf{r}_j(t) - \mathbf{r}_i(t)||^2}}$$

$$a_{ij} = ph\left(\frac{||\mathbf{r}_j(t) - \mathbf{r}_i(t)||_\sigma}{||r||_\sigma}\right) \; in \; [0, \; 1]$$

$$ph(x) = \begin{cases} 1 & x \in [0, \; h) \\ \frac{1}{2}\left[1 + \cos\left(\pi\frac{x - h}{1 - h}\right)\right] & x \in [h, \; 1] \\ 0 & \text{otherwise.} \end{cases}$$

$$f_i^\gamma = -w_4\left(\mathbf{r}_i(t) - \mathbf{r}_*(t)\right) - w_5\left(\mathbf{v}_i(t) - \mathbf{v}_*(t)\right)$$

Where $w_4, \; w_5 > 0, \; h \in (0, \; 1)$

| Arkin and Balch (1999) | | See Section 4.1.6 |

| Fine and Shell (2011) | $\mathbf{I} = \text{DistanceBased}_i(L^2, \mathbf{R}_{[0, \; \beta]})$ | $\mathbf{r}_i(t + \Delta t) = \begin{cases} \mathbf{r}_i(t) + \hat{\mathbf{e}}(t - \Delta t) & \text{if } |\mathbf{I}| = 0 \\ \mathbf{r}_i(t) + \hat{\mathbf{d}}_{i\mathbf{I}_0}(t) & \text{if } |\mathbf{I}| = 1 \text{ and } L^2(\mathbf{r}_i(t), \mathbf{r}_{\mathbf{I}_0}(t)) > r_\rho \\ \mathbf{r}_i(t) & \text{if } |\mathbf{I}| = 1 \text{ and } L^2(\mathbf{r}_i(t), \mathbf{r}_{\mathbf{I}_0}(t)) \le r_\rho \\ \mathbf{r}_i(t) & \text{if } \nexists k \in \mathbf{I} \text{ where } L^2(\mathbf{r}_i(t), \mathbf{r}_{\mathbf{I}_k}(t)) > w_1 \\ \mathbf{r}_i(t) + \hat{\mathbf{e}}(t) & \text{if } L^2(\mathbf{r}_i(t), \mathbf{r}_{a1}(t)) > r_\rho \\ \perp(\mathbf{r}_i(t) + \hat{\mathbf{e}}(t)) & \text{otherwise.} \end{cases}$ |

$$\mathbf{e}(t) = (\hat{\mathbf{d}}_{a2a1}(t) + \mathbf{r}_{a2}(t)) - \mathbf{r}_i(t)$$

$$a1 = \min(\forall k \in \mathbf{I}, \; L^2(\mathbf{r}_i(t), \; \mathbf{r}_k(t)))$$

$$a2 = \min(\forall k \in \mathbf{I}, \; L^2(\mathbf{r}_{a1}(t), \; \mathbf{r}_k(t)))$$

| Hauert et al (2011)[‖] | | $\theta_i(t + \Delta t) = \theta_i(t) - w_1\hat{\mathbf{s}}_i(t) + w_2\hat{\mathbf{a}}_i(t) + w_3\hat{\mathbf{c}}_i(t) + w_4\mathbf{d}_{i*}(t)$ |
| | $\mathbf{I} = \text{DistanceBased}(L^2, \mathbf{R}_{[0, \; \beta]})$ | |

$$\mathbf{s}_i(t) = \frac{\sum\limits_{j \in \mathbf{I}}\frac{1}{\mathbf{r}_i(t) - \mathbf{r}_j(t)}}{|\mathbf{I}|}$$

$$\mathbf{a}_i(t) = \frac{\sum\limits_{j \in \mathbf{I}}\mathbf{v}_j(t)}{|\mathbf{I}|}$$

$$\mathbf{c}_i(t) = \frac{\sum\limits_{j \in \mathbf{I}}\mathbf{r}_j(t)}{|\mathbf{I}|}$$

Table 4: The translation of the neighbor selection and motion computation stages from the selected flocking motion models. Together, these two stages constitute the motion rule, which is the primary focus of the vast majority of the literature. The motion rules presented in this table have be translated into the common notation (Table 1) presented earlier in this study. The neighbor selection column details the required perception functions (Table 2) along with what set(s) of neighbors will be considered. The motion computation column details the low-level control law, or algorithm, which computes the next motion of the flock member.

[‖] Uses a fixed velocity.

4.1 Literature Omitted from Table 4

Not all of the models from the selected literature can be easily formalized in our proposed framework (Matarić, 1993; Kelly and Keating, 1996; Reynolds, 1987; Arkin and Balch, 1999; Gueron et al, 1996), which have been labeled as "See Section 4.X.Y" in Table 4. The most common reason (found in all omitted works except for Gueron et al (1996)) for omission is ambiguity in the details of the low-level control law used to produce the flocking motions. Gueron et al (1996) is a unique case in which the motion rule is completely and precisely described, but the model is so complex, it does not lend it self to being formalized.

4.1.1 The Dynamics of Herds: From Individuals to Aggregations (Gueron et al, 1996)

In Gueron et al (1996), the authors presented the motion rule in enough detail where formalization is possible, but the rule is extremely verbose, which made it impractical to completely formalize the model in Table 4. Table 4 shows the formalization of one of the four spatial regions (*e.g.*, attraction, repulsion) which affects the flock member's motion. As shown in Table 4, the high number of discrete conditions in the motion computation stage requires a high number of perception functions in the neighbor selection stage. This suggests some care should be taken when designing the neighbor selection and motion computation stages.

4.1.2 Self-Organized Pedestrian Crowd Dynamics: Experiments, Simulations, and Design Solutions (Helbing et al, 2005)

Helbing et al (2005) presents a motion model based on the social force model presented in Helbing and Molnár (1995). This model, along with other similar models (Moussaïd et al, 2009), could be formalized into the presented notation if a complete and precise description of all of the forces were given. Specifically in Helbing et al (2005), the authors present the force $\mathbf{f}_{\alpha i}(\mathbf{r}_\alpha, \mathbf{r}_i, t)$ but only define the force as "attraction effects". Furthermore, the term $\xi_\alpha(t)$ is defined to be a "fluctuation term [that] reflects random behavioral variations". $\xi_\alpha(t)$ is clearly a noise term but the authors do not sufficiently define the properties of this term.

4.1.3 Designing Emergent Behaviors: From Local Interactions to Collective Intelligence (Matarić, 1993)

In Matarić (1993), the flocking motions are created from multiple behaviors, such as *Follow*, *Avoid*, *Aggregate*, and *Disperse*. Within the descriptions of each behavior there are ambiguities which make it difficult to formalize the behaviors. For example, the avoidance behavior in Matarić (1993) is composed of two types of avoidance; environmental and member. Within the environmental avoidance computation, there are ambiguous phrases such as "backup and turn", "turn right, go.", and "if an [obstacle] is on the right". It is difficult to translate these phrases into our proposed framework and notation without making certain decisions which could skew the original model design.

Another difficulty with the formalization of the avoidance behavior is the dynamics between the two types of avoidance. Both avoidance methods have the similar statement of "If an [obstacle/flock member] is on right...". If an obstacle is on the left and a flock member is on the right, it is unclear what the resulting behavior should be according to the description given in (Matarić, 1993). Similar ambiguities can be found in the other behaviors and the dynamics between them.

Even without these ambiguities, it would be difficult to recreate the motion model presented in Matarić (1993). The presentation of the behaviors uses phrases such as "backup" and "turn right". If we where to formalize these behaviors, the majority of the terms would be parameters and not calculations based on input. This is not a problem when it comes to validating the model which produces flocking motions, but it does make it difficult to reproduce the work for further study and comparisons.

Version	Neighbor Selection	Motion Computation
Reynolds (2004)	$\mathbf{I} = \mathrm{DistanceBased}(L^2, \mathbf{R}_{[0,\ \beta]})$	$\mathbf{v}_i(t) \;=\; w_1\hat{\mathbf{s}}_i(t) + w_2\hat{\mathbf{a}}_i(t) + w_3\hat{\mathbf{c}}_i(t)$ $\mathbf{s}_i(t) \;=\; \dfrac{\displaystyle\sum_{j\in\mathbf{I}} \dfrac{\mathbf{r}_i(t) - \mathbf{r}_j(t)}{-\delta_2(\mathbf{r}_i(t),\ \mathbf{r}_j(t))^2}}{|\mathbf{I}|}$ $\mathbf{a}_i(t) \;=\; \dfrac{\displaystyle\sum_{j\in\mathbf{I}} \theta_j(t)}{|\mathbf{I}|} \;-\; \theta_i(t)$ $\mathbf{c}_i(t) \;=\; \dfrac{\displaystyle\sum_{j\in\mathbf{I}} \mathbf{r}_j(t)}{|\mathbf{I}|} \;-\; \mathbf{r}_i(t)$

$$\mathbf{v}_i(t) \;=\; w_1\hat{\mathbf{s}}_i(t) + w_2\hat{\mathbf{a}}_i(t) + w_3\hat{\mathbf{c}}_i(t)$$

Kline (1996)

$\mathbf{I} = \mathrm{DistanceBased}(L^2, \mathbf{R}_{[0,\ \beta]})$

$$\mathbf{s}_i(t) = \begin{cases} 0 & \text{if } k = \emptyset, \\[2mm] \delta_2(\mathbf{r}_i(t),\ \mathbf{r}_j(t)) & \text{otherwise.} \end{cases}$$

$\mathbf{I} = \mathrm{DistanceBased}(L^2, \mathbf{R}_{[0,\ \alpha]}) \cap \mathrm{Nearest}()$

$$\mathbf{a}_i(t) = \begin{cases} 0 & \text{if } k = \emptyset, \\[2mm] \mathbf{a}_i x(t) = \begin{cases} \mathbf{a}_i x(t) - w_5 & \text{if } (\mathbf{r}_k(t) - \mathbf{r}_i(t))x \;<\; w_4, \\ \mathbf{a}_i x(t) + w_6 & \text{otherwise.} \end{cases} & \\[4mm] \mathbf{a}_i y(t) = \begin{cases} \mathbf{a}_i y(t) - w_5 & \text{if } (\mathbf{r}_k(t) - \mathbf{r}_i(t))y \;<\; w_4, \\ \mathbf{a}_i y(t) + w_6 & \text{otherwise.} \end{cases} & \text{otherwise.} \\[4mm] \mathbf{a}_i z(t) = \begin{cases} \mathbf{a}_i z(t) - w_5 & \text{if } (\mathbf{r}_k(t) - \mathbf{r}_i(t))z \;<\; w_4, \\ \mathbf{a}_i z(t) + w_6 & \text{otherwise.} \end{cases} & \end{cases}$$

$$\mathbf{c}_i(t) \;=\; \dfrac{\displaystyle\sum_{j\in\mathbf{I}} \mathbf{r}_j(t)}{|\mathbf{I}|} \;-\; \mathbf{r}_i(t)$$

Table 5: Both entries in this table are possible implementations of the motion model in Reynolds (1987). The two implementations only vary slightly, however, the differences have effects on the complexity of the flock members and the underlying causes of flocking motions.

4.1.4 Flocks, herds and schools: A distributed behavioral model (Reynolds, 1987)

As in the previous studies discussed, Reynolds (1987) is difficult to formalize without introducing bias. Even though the three rules for flocking presented in Reynolds (1987) are highly cited in the robotics literature, the details of the three rules are ambiguous. Again, the problem lies in the details of the low-level behaviors and vague descriptions of the various parameters needed.

Table 5 shows two possible formalizations of the flocking motion model in Reynolds (1987). Both Reynolds (2004) and Kline (1996) were formalized using actual software implementations of Reynolds proposed model. Even though these formalizations are similar and produce similar flocking motions, there are a few key differences.

First, Kline (1996) only considers the closest flock member within a given radius when computing $\mathbf{a}_i(t)$ and $\mathbf{s}_i(t)$ where Reynolds (2004) considers all flock members within the sensing radius. The computational differences between these two approaches have a direct affect on the neighbor selection function. Please note that in the formalization of the neighbor selection function in Kline (1996), we had to include the union of DistanceBased($\cdot$) $\cap$ Nearest() in order to select the nearest neighbor, which is explicitly used in the motion computation stage.

21

4.1.5 On Flocking By The Fusion Of Sonar And Active Infrared Sensors (Kelly and Keating, 1996)

Similar to Matarić (1993), Kelly and Keating (1996) has ambiguous behavior descriptions, which makes it difficult to formalize into our unifying framework. In Kelly and Keating (1996) the authors present the dynamics between the different behaviors as a hierarchy but does not fully present the underling behaviors. The authors use phrases such as "try to maintain position" and "speed up", which are difficult to formalize without making assumptions on the author's intent.

4.1.6 Behavior-based Formation Control for Multi-robot Teams (Arkin and Balch, 1999)

This work is a prime example of how researchers should report all of the various parameters and gains used in the validation process. In addition to there being no ambiguities in respect to the parameters and gains, the authors list the actual values used, thus allowing for repeatability. However, we were unable to formalize this work due to some ambiguity in the presentation of the primitive schemas used. For example, the **Move-to-goal** schema states "Attract to goal with variable gain. Set high when heading for goal." Even though, it seems straight forward to formulate this schema, we do not attempt to formalize this work for the same reasons as in Section 4.1.5.

5 Validation Methods

In this section we analyze the various methods of motion model validation found in the selected literature. Table 6 is a review of the validation choices (*e.g.*, synchronous versus asynchronous flock members) and the various validation methods (*e.g.*, computer simulations versus physical implementations). The six attributes (validation method, synchrony, neighbor's position, neighbor's velocity, neighbor's orientation, flock's environment) highlighted in Table 6 give an adequate description of how current models are being validated. The information provided in Table 6 also affords us the ability to gain insight into which models may be more effective in a real world situation. In other words, if a motion model is only validated using global information in an obstacle free environment, this model may not produce flocking motions when introduced in a more realistic environment. Additionally, Table 6 can be used to cross-check the assumptions made in the design of a given model (*e.g.*, a model design for local information should be validated with local information).

Paper	Validation Method	Synchrony	Position	Velocity	Orientation	Environment
Viscido et al (2002)(SNN/HA/LCH)	$\mathcal{S}$	√	Global	–	–	Bounded Free Space
Warburton and Lazarus (1991)	$\mathcal{S}$	√	Global	Global	Global	Unbounded Free Space
Conradt et al (2009)	$\mathcal{S}$	√	Global	Global	Global	Unbounded Free Space
Codling et al (2007)	$\mathcal{S}$	√	Global	Global	Global	Unbounded Free Space
Gueron et al (1996)	$\mathcal{S}$	√	Local	Global	Global	Unbounded Free Space
Couzin et al (2005)	$\mathcal{S}$	√	Global	Global	Global	Unbounded Free Space
Lopez et al (2012)	$\mathcal{S}$	√	Global	Global	Global	Unbounded Free Space
Huth and Wissel (1992)	$\mathcal{S}$	√	Global	Global	Global	Unbounded Free Space
Hamilton (1971)	$\mathcal{S}$	√	Global	–	–	Unbounded Free Space
Vicsek et al (1995)	$\mathcal{S}$	√	Global	Constant	Global	Periodic Free Space
Dong (2012)	$\mathcal{M}$	√	Global	Global	–	Unbounded Free Space
Smith and Martin (2009)	$\mathcal{S}$	√	Global	Constant	Global	Periodic Free Space
Shimoyama et al (1996)	$\mathcal{S}$	√	Global	Global	Global	Unbounded Free Space
Czirók et al (1997)	$\mathcal{S}$	√	Global	Global	Global	Periodic Free Space

Paper	Validation Method	Synchrony	Position	Velocity	Orientation	Environment
Szabó et al (2008)	$\mathcal{S}$	√	Global	Global	Global	Periodic Free Space
Szabó et al (2009)	$\mathcal{S}$	√	Global	Global	Global	Periodic Free Space
Levine et al (2000)	$\mathcal{S}$	–	Global	Constant	Global	Periodic Free Space
Toner and Tu (1998)	$\mathcal{M}$	√	Global	Constant	Global	Bounded Free Space
Grégoire et al (2003)	$\mathcal{S}$	√	Global	Constant	Global	Periodic Free Space
Camperi et al (2012)	$\mathcal{S}$	√	Global	Global	–	Unbounded Free Space
Helbing et al (2000)	$\mathcal{SP}^{**}$	–	Global	Global	–	Bounded Obstacles
Matarić (1993)	$\mathcal{P}$	–	Local	–	Local	Bounded Free Space
Reynolds (1987)	$\mathcal{S}$	√	Global	Global	Global	Unbounded Obstacles
Kelly and Keating (1996)	$\mathcal{P}$	–	Local	Inferred	Inferred	Bounded Free Space
Turgut et al (2008)	$\mathcal{SP}$	–	Global	–	Local	Bounded Free Space
Gökçe and Şahin (2009)	$\mathcal{SP}$	–	Global	–	Local	Bounded Free Space
Tanner et al (2003a)	$\mathcal{MS}$	√	Global	Global	Global	Bounded Free Space
Tanner et al (2003b)	$\mathcal{MS}$	√	Global	Global	Global	Unbounded Free Space
Jadbabaie et al (2002)	$\mathcal{M}$	√	Global	Constant	Global	–
Gazi and Passino (2005)	$\mathcal{MS}$	–	Global	Global	Global	Unbounded Free Space
Gazi and Passino (2003)	$\mathcal{M}$	√	Global	–	–	–
Olfati-Saber (2006)	$\mathcal{MS}$	√	Global	Global	Global	Unbounded Obstacles
Arkin and Balch (1999)	$\mathcal{SP}$	–	Global	–	–	Unbounded Obstacles
Fine and Shell (2011)	$\mathcal{P}$	–	Local	–	–	Bounded Free Space
Hauert et al (2011)	$\mathcal{SP}$	–	Local	Local	–	Unbounded Free Space

Legend: $\mathcal{M}$: Mathematical $\mathcal{S}$: Simulation $\mathcal{P}$: Physical

Table 6: Details the validation methods chosen for the selected motion models. The form of validation employed varies significantly within the publications.

5.1 Definitions of Attributes for Table 6

5.1.1 Validation Method

The validation method attribute details what combination of the possible methods where used to validate the motion model and characteristics of the flocking motion produced by the model. The methods seen in the selected literature are mathematical verification ($\mathcal{M}$) of flocking motion characteristics (*e.g.*, flock stability), computer simulation ($\mathcal{S}$), and the use of physical implementations ($\mathcal{P}$). Both computer simulations and physical implementations rely on implementing the motion model and studying the exhibited motions of the flock. Literature that use mathematical methods typically prove the existence of various group-level characteristics (*e.g.*, does the flock converge to a stable formation or does a phase shift occur). The difference between validation (computer simulations and physical implementations) and verification (mathematical) lies in the scope of the chosen method. Validation methods consider the overall motions produced by the flock, where verification methods consider particular aspects of the flocking motions (*e.g.*, convergence and flock stability).

5.1.2 Synchrony

Synchrony defines whether or not flock members sense, compute, and act in unison. If one flock member executes any of the five stages at a different time or frequency, validation is said to be asynchronous. The key distinction between synchronous and asynchronous validation is that if the model is asynchronous, then we cannot assume the sets $\mathbf{A}_i(t)$ and $\mathbf{A}_j(t)$ are identical because the sensing of the flock members could have been executed at different times.

**The physical agent were human participants.

5.1.3 Position/Velocity/Orientation

The position, velocity, and orientation attributes describe the way in which the required information (see Table 3) is actually sensed. The three sensing methods found in the literature are global (Global), local (Local), or inferred from other information (Inferred). In the selected publications global information is either sensed by an overhead sensor or by an oracle that maintains the required information. Local information is gathered by the flock member via sensors or member-to-member communication. Inferred information can be gathered in two ways, (1) by using two distinct types of information (*e.g.*, speed and orientation can yield velocity) or (2) by associating data from multiple sensing iterations (*e.g.*, displacement of position over time can yield velocity).

5.1.4 Environment

The environment attribute describes the type of environment considered in the validation of the proposed model. The environment can be any combination of bounded, unbounded, or periodic space, with or without obstacles. For all of the literature reviewed in this study, all of the models where designed with regards to a particular environment. There are no cases in the selected literature where a motion model was tested in an environment it was not designed for.

5.2 Observations from Table 6

5.2.1 Validation Methods

The most commonly seen form of validation is the use of computer simulations which is then followed by physical implementations. It is important to note there are various degrees of simulation validation (*e.g.*, physics-based, sensor-based, etc.), which are not covered in this meta-study. Mathematical verifications are typically used to show that certain known flock properties hold given a particular model. In Tanner et al (2003a) and Tanner et al (2003b) the authors uses graph theory to show the flock members maintain a stable flock (*i.e.*, all flock members maintain common velocities and avoid collisions with other flock members). However, mathematical verifications are also used to explore certain properties of the flock, such as phase shifts in Mikhailov and Zanette (1999)[††]. The most complete validation treatment of a proposed microscopic model found in the selected literature was done in Lindhé et al (2005). This work uses all three methods of validation covered in this meta-study.

5.2.2 Synchrony

From Table 6 we observe that ten of the publications consider asynchronous flock members. Additionally, we noticed that all of the publications that consider asynchronous flock members are from the control and robotics literature. The low number of asynchronous flocks is surprising because we know from Şamiloğlu et al (2006) that asynchronism can have negative impacts on the exhibited flocking motions.

5.2.3 Position/Velocity/Orientation

All of the motion models from the selected literature only require local information (see formalization of motion rules in Table 4); however, many of the studies use global information for the validation. This disconnect between design and validation leaves room for subtle assumptions which could affect the flock's overall motion; such as the effects of occlusions. Physical flock members using local sensing may not be able to detect the same set of neighbors as a global sensor due to occlusions from flock members and environmental obstacles (*i.e.* the set $\mathbf{D}_i(t)$ using a global sensor may differ from the same set sensed via a local sensor). For example, if there are three collinear flock members, a member on the end of that line may only be able to detect one neighbor using local sensing rather than both neighbors. This issue becomes apparent when considering the Hamiltonian (HA) motion rule from Viscido et al (2002).

[††]Phase shifts refer to the moments in time when a flock becomes or discontinues being a flock.

If the motion computation stage presented in Viscido et al (2002) uses local information, the motion computation stage becomes undefined in some cases. Since the HA motion model was validated using global information (with the absence of occlusions), the motion computation stage does not treat cases where only one neighbor is detected; thus presenting a potential problem when local information is used. For a more detailed treatment of the HA motion rule please see Viscido et al (2002) and Fine and Shell (2011). There is not an instance in the selected literature where the effects of occlusions are properly treated or studied.

Velocity information is, arguably, the most complex type of information used in the literature; being a combination of a member's speed and orientation. Typically, flock members need to employ extra strategies and information in order to attain the velocity of its neighbors; either through communication and identification, or data association. There are only a few works in the selected literature (Kelly and Keating, 1996; Vicsek et al, 1995; Czirók et al, 1997) which describe how the velocity information is attained. The other literature simply states that the motion computation stage uses velocity information.

5.2.4   Environment

The environment used for physical and simulated validation is an important aspect to consider when planning to implement a particular approach on a physical system. None of the selected literature has investigated the effects that the environment has on the motions of the flock, with very few studies considering environmental obstacles (Reynolds, 1987; Olfati-Saber, 2006; Lindhé et al, 2005; Arkin and Balch, 1999). Investigations which do consider environmental obstacles typically select environments with few obstacles which are widely spaced from each other. Olfati-Saber (2006) is one of the more complete treatments with respect to the effects the environment has on the flock's motions.

6   Discussion

This meta-study presented three types of tools (data-flow model, two taxonomies, and a notation/formalization) to assist in the reduction of incompleteness and imprecision in microscopic flocking motion models. The proposed DT along with the two taxonomies and the notation/formalization allow for better understanding and comparison of the current literature on flocking motion models; however, there do exist some cases and investigations in which the current tool-set does not work as well as it could. Through the exploration of these outlying cases, we have highlighted future avenues of research that could prove beneficial to the overall understanding of the flocking phenomenon.

Case-study: Fine and Shell (2011)

The flocking motion model presented in Fine and Shell (2011) is the Hamiltonian (HA) motion model first presented in Viscido et al (2002). In Fine and Shell (2011) we present the HA motion model using the tool-set presented in this meta-study in-mind. By translating the HA motion model from Viscido et al (2002) into the notation presented here, we showed the existence of subtle inconsistencies and unrealistic assumptions. Table 4 shows the difference in completeness for both the motion models presented in Viscido et al (2002) and Fine and Shell (2011). Furthermore, by applying the DT to the model in Viscido et al (2002), we were able to show that the exhibited motions of any motion model are independent of the perception function used (under certain parameter constraints).

Collision Avoidance in the Data-flow Template

Although our model is general, it imposes enough constraints to serve as a constructive guide toward consistent, complete, and precise descriptions of flocking motion models. One consideration not covered in the current DT is the motion computation stage's ability to handle collisions. The majority of the literature states that flock members avoid collisions with the environment, but the publications rarely describe the process/computations required to perform such motions (*e.g.*, how does the flock member detect the environment, which parts (objects) are used in the computation stage, and/or how do the avoidance computations affect the motion computations). To allow for a better understanding of the collision avoidance capabilities,

we must modify the DT as presented in Figure 1; which only defines the motion model when there are no obstacles in the environment.

To address collision avoidance in the DT we introduced the modified DT in Figure 3, which shows the addition of a sixth and seventh stage (obstacle detection and obstacle selection). These two additional stages are executed in parallel with the flock member detection and neighbor selection stages, respectively. The obstacle detection stage takes the same input as the flock member detection stage (input from the flock member's sensors) and produces a set of obstacles. This set can be passed to the obstacle selection stage, which outputs a subset of detected obstacles for the motion computation stage, or the set can be paired with the flock members in the set $\mathbf{D}_i(t)$ to generate virtual flock members (see ① in Figure 3).
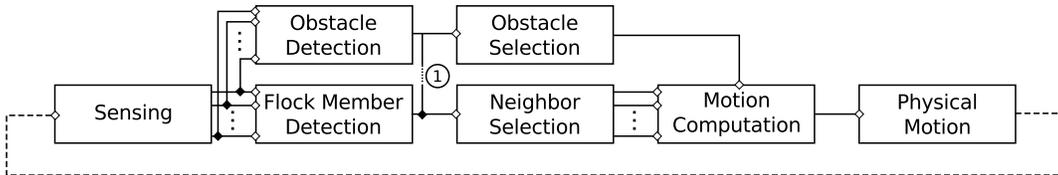
Figure 3: The modified DT that includes the obstacle detection and obstacle selection stages. These two stages allow for the addition of obstacle avoidance behaviors and the generation of virtual flock members. The connection at ① can be treated as a place holder for adding a method that generates the required virtual flock members, as in Olfati-Saber (2006).

Another aspect of flock member collisions, either with other members or the environment, not covered in the current body of the flocking literature is the ability to use collisions as input to the motion model. There have been studies in biology that have suggested that certain ants use information form ant-to-ant collisions in order to adjust their behaviors. Even though these studies focus on task switching, it is plausible to assume certain flocking motion models may use collisions as input to the motion computation stage (*e.g.*, flock members using collisions to navigate the environment). It would be interesting to see if it is possible to create flocking motions with only using information sensed from direct contact and how that might affect the common assumptions made in the flocking motion literature.

Implicit Assumptions in Flock Member Detection

The vast majority of the flocking literature to date has made the implicit assumption of agent-based detection. A flock member detection stage that uses agent-based detection takes all of the sensed information from a single flock member and reduces it to a single description of that particular flock member. For example, if a model requires the position of the k neighbors, then the flock member detection stage would output (to the neighbor selection stage) k position values. This would be the case even if the sensor took multiple readings from the same neighbor (*e.g.*, a neighbor could take-up multiple laser-range finder readings due to its physical size). Fine and Shell (2011) show that the implicit assumption of agent-based detection is not necessary for the production of flocking motions and introduces flocking motion through the use of sensor-based detection, which does not reduce the sensor information to a single description. Similarly, Fine and Shell (2012) show that flocking motions can be produced by simply detecting groups of flock members, instead of detecting the individual members. The work done in Fine and Shell (2011) and Fine and Shell (2012) show that implicit assumptions in the flock member detection stage can have negative affects on both the exhibited flocking motion and on the completeness of the flocking motion model; thus care must be taken when determining the model's assumptions.

Neighbor Identification

In theory, identification can prove useful for maintaining the flock's structure and determining a flock member's membership within the flock (*e.g.*, is the flock member a leader or follower). Unfortunately, obtaining identification information in practice (on a physical implementation) is difficult and error prone. On a physical system there are three common ways to obtain the required identification information; (1)

member-to-member communication, (2) direct sensing (*e.g.*, flock members can identify the 'color' of its neighbor), and/or (3) association. To date, it is not clear if biological flock members utilize identification, but Occam's razor would suggest that identification would not be required since flocking motions can be produced without it. Clearly, further investigations are needed in order to understand the role, if any, identification plays in the production of flocking motion.

Association

Association is the ability for a flock member to pair sensor information from two consecutive sensor readings. Association can be used in three ways, (1) as a standalone part of a given model (i.e., using prior information as input to the motion model), (2) inferring information from multiple readings (*e.g.*, using a flock member's position displacement to determine that member's velocity), and/or (3) using association to aid in identification (*e.g.*, if the flock member knows the starting positions of its neighbors, the flock member could use association to keep track of its neighbors' IDs). It is important to note, association and identification are distinct attributes that are independent of each other (*i.e.*, it is possible to use any combination of these attributes). We have omitted the association column from Table 3 due to the lack of proper presentation in the selected literature.

Data-Centric Approach to Determining Parameter Values

As we see in Kline (1996); Olfati-Saber (2006); Gökçe and Şahin (2009); Shimoyama et al (1996) many publications present motion models that contain many different parameters or gains. Very few works, if any, describe how the parameters for the motion model's validation where selected, where some works (Vicsek et al, 1995) validate the model over a range of parameter settings. Even in the investigations that study a range of parameter values, the justification of the chosen values remains unclear. The majority of the time the values are artificially 'tuned' until the desired flocking motions are produced from the given model. Useful information about the effects of parameter values may be found if we determine the parameter values from biological flocks (Moussaïd et al, 2009; Lopez et al, 2012; Lukeman et al, 2010). For example, the repulsion radius of the flock members could be determined by analyzing the average distance maintained between the members of the biological flock.

Flocking Motion is Independent to Information Types

Observing Table 3, we can seen that almost every possible combination of required information was used in the studied motion models. If we now only consider the information required by the motion computation stage, the number of combinations seen in the literature increases. From this, it is reasonable to assume that flocking motion models do not require a specific type of information. Further support for this observation can be found in Fine and Shell (2012) with the introduction of information-abstracted flocking. Fine and Shell (2012) show the existence of a flocking motion model that is structured in away as to allow for the use of different combinations of information without modifying the motion model.

Physical Implementations in Three Dimensions

The flocking phenomenon is seen all throughout nature in both two-dimensional (*e.g.*, sheep and cattle) and three-dimensional animals (*e.g.*, fish and birds). With that said, the vast majority of physical implementations of motion models only exist on two-dimensional wheeled robotic platforms. Recently, however, there has been increased research interest in implementing flocking motion on flying robots (Hauert et al, 2011). Hauert et al (2011) implements the motion model first presented by Reynolds (1987) on fixed-wing flying robots.

6.1   Recommendations from the Tool-Set

The goal of the following recommendations are to help outline a framework/style for the presentation and design of future microscopic flocking motion models. All of the recommendations have resulted from applying

the previously presented tools to the selected publications. We feel that if these recommendations are followed for future publications, the overall understanding of the flocking problem will be enhanced.

Data-flow Template (see Section 2)

1. The available and type of raw sensor information in the sensing stage affects all other aspects of the flocking motion model. Therefore, the sensing stage should explicitly list how all of the required information (*e.g.*, position, velocity, identification, etc.) is sensed from the flock member's environment.

2. We have seen in multiple studies (Vicsek et al, 1995) that not all required information is used in the motion computation stage. For those works, it remains unclear if that information is a part of the motion model, or if that information is simulating a sensor or flock member limitation. Therefore, the purpose and use of all required information should be clearly described.

3. Specific sensing attributes and limitations (*e.g.*, agent-based detection) may affect the exhibited flocking motions and/or the design of the motion model. Therefore, the sensing stage and the flock member detection stage should present any and all assumptions made.

4. As we saw in the presentation of the Viscido et al (2002) and Reynolds (1987) motion models, there is a possibility for multiple interpretations of a given model. To help reduce ambiguities, algorithmic presentations of motion computation stages should be preferred over prose descriptions.

5. The vast majority of the flocking motion literature treats/presents multiple stages as one stage (typically the neighbor selection and motion computation stages), which could lead to incompleteness and/or imprecision. Therefore, we recommend that all five DT stages should be logically separated and should also be treated/presented separately.

Design and Validation Taxonomies (see Sections 3 and 5 respectively)

1. Ambiguities and/or omissions of important information in the current motion models make it difficult to compare works across the literature. Therefore, future publication should explicitly state what attributes and assumptions the motion model requires and utilizes, respectively.

2. To better accommodate the implementation of flocking motion models on physical systems (robotic or biological), continuous time models should be preferred over discrete time models.

3. In some of the literature, the motion model was design for local information, but when the model was validated the flock members had access to global information. This inconsistency in the motion model validation could lead to models which are difficult to implement on a physical system. Therefore, the method in which the required information is sensed should reflect the type of information required by motion computation stage (*i.e.*, globally versus locally sensed).

4. To better simulate real-world situations, future motion models should be validated using asynchronous flock members (when applicable).

5. The phenomenon of flocking is created through the interaction of many distributed individual flock members. Therefore, motion models should only use locally sensed information when investigating the flocking phenomenon.

6. Due to flocks existing and operating in real-world environments, future investigations should validate motion models in obstacle filled environments (similar to environments biological flocks may encounter) or detail the assumptions that make this unnecessary or potentially detrimental to the model.

7   Conclusion

In this meta-study we have explored the current state of flocking motion literature that focuses on microscopic flocking motion models. Using the three presented tools (data-flow template, two taxonomies, and a formalization/notation) we have identified and detailed the commonly seen designs and assumptions in the current motion models. Through the use of our DT we have identified that there are at least five critical aspects of the flocking problem. We have shown that failure to properly treat all of the five stages of the DT could lead to incompleteness and/or imprecision in the presentation of the motion model. To demonstrate this, we have detailed examples were adhering to the DT, leads to a more complete and precise understanding of the model (Section 2.2). Additionally, through analysis of the selected publications using the two taxonomies (Tables 3 and 6), we have identified the common assumptions made in the literature, and have shown that the majority of the investigations of microscopic motion models have the same underlying aim. Therefore, in an attempt to increase the breadth of flocking motion research, we make the following recommendations for future research investigations.

7.1   Directions for Future Investigations

1. The vast majority of the literature uses agent-based flock member detection, therefore we suggest that future robotic investigations explore other detection methods, such as sensor-based detection.

2. As mentioned in Section 3, there are other possible definitions of group composition; thus, future studies could consider the effects the various definitions have on the motion model and the exhibited motions of the flock.

3. Future investigations could explore what flocking motions are afforded by using collisions as input to the motion computation stage.

4. The vast majority of the flocking motion models require position information from the flock member's neighbors. Therefore, future investigations could explore the production of flocking motions without the use of position information.

5. It is reasonable to assume that the output of the motion computation stage will affect the input of the sensing stage (as seen in Figure 1). Future investigations could explore this connection in depth and study various aspects of a physical flock, such as occlusions.

References

Albi G, Pareschi L (2012) Modeling self-organized systems interacting with few individuals: from microscopic to macroscopic dynamics. ArXiv e-prints

Aoki I (1984) Internal Dynamics of Fish Schools in Relation To Inter-Fish Distance. Bulletin of the Japanese Society of Scientific Fisheries 48(3):1081–1088

Arkin RC, Balch T (1999) Behavior-based Formation Control for Multi-robot Teams. IEEE Transactions on Robotics and Automation 14(6):926–939

Babak P, Magnsson KG, Sigurdsson S (2004) Dynamics of group formation in collective motion of organisms. Mathematical Medicine and Biology 21(4):269–292

Ballerini M, Cabibbo N, Candelier R, Cavagna A, Cisbani E, Giardina I, Lecomte V, Orlandi A, Parisi G, Procaccini A, Viale M, Zdravkovic V (2008) Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study. Proceedings of the National Academy of Sciences 105(4):1232–1237

Barbosa A (1995) Foraging Strategies and Their Influence on Scanning and Flocking Behaviour of Waders. Journal of Avian Biology 26(3):182–186

Bazazi S, Buhl J, Hale JJ, Anstey ML, Sword GA, Simpson SJ, Couzin ID (2008) Collective Motion and Cannibalism in Locust Migratory Bands. Current Biology 18(10):735–739

Bender JG, Fenton RE (1970) On the Flow Capacity of Automated Highways. Transportation Science 4(1):52–63

Blomqvist O, Bremberg S, Zauer R (2012) Mathematical modeling of flocking behavior.

Bode NWF, Franks DW, Wood AJ (2011) Limited interactions in flocks: relating model simulations to empirical data. Journal of The Royal Society Interface 8(55):301–304

Camperi M, Cavagna A, Giardina I, Parisi G, Silvestri E (2012) Spatially balanced topological interaction grants optimal cohesion in flocking models. Interface Focus 2(6):715–725

Cavagna A, Cimarelli A, Giardina I, Parisi G, Santagati R, Stefanini F, Tavarone R (2010) From empirical data to inter-individual interactions: Unveiling the rules of collective animal behavior. Mathematical Models and Methods in Applied Sciences 20:1491–1510

Cavagna A, Giardina I, Ginelli F (2012) Boundary information inflow enhances correlation in flocking. ArXiv e-prints

Clark PJ, Evans FC (1954) Distance to Nearest Neighbor as a Measure of Spatial Relationships in Populations. Ecology 35(4):445–453

Codling EA, Pitchford JW, Simpson SD (2007) Group navigation and the "many-wrongs principle" in models of animal movement. Ecology 88(7):1864–1870

Conradt L, Krause J, Couzin ID, Roper TJ (2009) "Leading According to Need" in Self-Organizing Groups. The American Naturalist 173(3):304–312

Couzin ID, Krause J, Franks NR, Levin SA (2005) Effective leadership and decision making in animal groups on the move. Nature 433(3):513–516

Şamiloğlu AT, Gazi V, Koku AB (2006) Effects of asynchronism and neighborhood size on clustering in self-propelled particle systems. In: Computer and Information Sciences ISCIS 2006, Lecture Notes in Computer Science, vol 4263, Springer Berlin / Heidelberg, pp 665–676

Czirók A, Stanley HE, Vicsek T (1997) Spontaneously ordered motion of self-propelled particles. Journal of Physics A: Mathematical and General 30(5):1375–1385

Dingle H, Drake AV (2007) What Is Migration? BioScience 57(2):113–121

Dong JG (2012) Flocking under hierarchical leadership with a free-will leader. International Journal of Robust and Nonlinear Control pp n/a–n/a

Edelstein-Keshet L (2001) Mathematical models of swarming and social aggregation. In: International Symposium on Nonlinear Theory and its Applications, Miyagi, Japan

Emlen, Jr JT (1952) Flocking Behavior in Birds. The Auk 69(2):160–170

Ferrante E, Turgut AE, Huepe C, Stranieri A, Pinciroli C, Dorigo M (2012) Self-organized flocking with a mobile robot swarm: a novel motion control method. Adaptive Behavior 20(6):460–477

Fine BT, Shell DA (2011) Flocking: don't need no stink'n robot recoginition. In: IEEE/RSJ International Conference on Robotics and Automation, San Francisco, CA, USA

Fine BT, Shell DA (2012) Examining the information requirements for flocking motion. In: 12th International Conference on Adaptive Behavior, Odense, Denmark

Gazi V, Passino KM (2003) Stability Analysis of Swarms. IEEE Transactions on Automatic Control 48(4):692–697

Gazi V, Passino KM (2005) Stability of a one-dimensional discrete-time asynchronous swarm. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 35(4):834–841

Giardina I (2008) Collective behavior in animal groups: Theoretical models and empirical studies. HFSP Journal 2(4):205–219

Ginelli F, Chaté H (2010) Relevance of metric-free interactions in flocking phenomena. Phys Rev Lett 105:168,103

Gökçe F, Şahin E (2009) To flock or not to flock: the pros and cons of flocking in long-range "migration" of mobile robot swarms. In: Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems (AAMAS '09), Budapest, Hungary, pp 65–72

Goldstone RL, Janssen MA (2005) Computational models of collective behavior. Trends in Cognitive Sciences 9(9):424–430

Grégoire G, Chaté H, Tuj Y (2003) Moving and staying together without a leader. Physica D: Nonlinear Phenomena 181(30-4):157–170

Gueron S, Levin SA, Rubenstein DI (1996) The Dynamics of Herds: From Individuals to Aggregations. Journal of Theoretical Biology 182(1):85–98

Hamilton WD (1971) Geometry for the Selfish Herd. Journal of Theoretical Biology 31(2):295–311

Hauert S, Leven S, Varga M, Ruini F, Cangelosi A, Zufferey JC, Floreano D (2011) Reynolds flocking in reality with fixed-wing robots: communication range vs. maximum turning rate. In: IEEE/RSJ International Conference on Robotics and Automation, San Francisco, CA, USA

Helbing D, Molnár P (1995) Social force model for pedestrian dynamics. Phys Rev E 51:4282–4286

Helbing D, Farkas I, Vicsek T (2000) Simulating dynamical features of escape panic. Nature 407:487–490

Helbing D, Buzna L, Johansson A, Werner T (2005) Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions. Transportation Science 39(1):1–24

Hildenbrandt H, Carere C, Hemelrijk C (2010) Self-organized aerial displays of thousands of starlings: a model. Behavioral Ecology 21(6):1349–1359

Huth A, Wissel C (1992) The simulation of the movement of fish schools. Journal of Theoretical Biology 156(3):365–385

Hutto RL (1988) Foraging Behavior Patterns Suggest a Possible Cost Associated with Participation in Mixed-Species Bird Flocks. Oikos 51(1):79–83

Ip GW, Chiu Cy, Wan C (2006) Birds of a feather and birds flocking together: Physical versus behavioral cues may lead to trait- versus goal-based group perception. Journal of personality and social psychology 90(3):368–368–381

Jadbabaie A, Lin J, Morse AS (2002) Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbor Rules. IEEE Transactions on Automatic Control 48(6):988–1001

James R, Bennett PG, Krause J (2004) Geometry for mutualistic and selfish herds: the limited domain of danger. Journal of Theoretical Biology 228(1):107–113

Kelly ID, Keating DA (1996) On Flocking By The Fusion Of Sonar And Active Infrared Sensors. In: Proceedings of the Conference on Mechatronics and Machine Vision in Practice, Guimarães, Portugal, vol 1, pp 14–17

Kline C (1996) C++ boids. http://www.behaviorworks.com/people/ckline/cornellwww/boid/boids.html, Last viewed in March 2012

Levine H, Rappel WJ, Cohen I (2000) Self-organization in systems of self-propelled particles. Physical Review E 63(1):017,101–017,104

Lindhé M, Ögren P, Johansson KH (2005) Flocking with Obstacle Avoidance: A New Distributed Coordination Algorithm Based on Voronoi Partitions. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'05), Barcelona, Spain, pp 1785–1790

Lopez U, Gautrais J, Couzin ID, Theraulaz G (2012) From behavioural analyses to models of collective motion in fish schools. Interface Focus

Lukeman R, Li YX, Edelstein-Keshet L (2010) Inferring individual rules from collective behavior. Proceedings of the National Academy of Sciences 107(28):12,576–12,580

Matarić MJ (1993) Designing Emergent Behaviors: From Local Interactions to Collective Intelligence. In: Proceedings of the Second International Conference on Simulation of Adaptive Behavior (SAB '93), Honolulu, Hawai, USA, pp 432–441

Mikhailov AS, Zanette DH (1999) Noise-induced breakdown of coherent collective motion in swarms. Physical Review E 60(4):4571–4575

Miki T, Nakamura T (2006) An effective simple shepherding algorithm suitable for implementation to a multi-mmobile robot system. In: Innovative Computing, Information and Control, 2006. ICICIC '06. First International Conference on, vol 3, pp 161 –165

Mogilner A, Edelstein-Keshet L (1999) A Non-Local Model for a Swarm. Journal of Mathematical Biology 38:534–570

Moussaïd M, Helbing D, Simon Garnier MC Anders Johansson, Theraulaz G (2009) Experimental study of the Behavioural mechanisms underlying self-organization in human crowds. Proceedings of the Royal Society B 276:2755–2762

Niizato T, Gunji YP (2011) Metrictopological interaction model of collective behavior. Ecological Modelling 222(17):3041 – 3049

Okubo A (1986) Dynamical aspects of animal grouping: Swarms, schools, flocks, and herds. Advances in Biophysics 22:1–94

Olfati-Saber R (2006) Flocking for Multi-agent Dynamic Systems: Algorithms and Theory. IEEE Transactions on Automatic Control 51(3):401–420

Parrish JK (1989) Re-examining the selfish herd: are central fish safer? Animal Behaviour 38(6):1048–1053

Parrish JK, Edelstein-Keshet L (1999) Complexity, Pattern, and Evolutionary Trade-Offs in Animal Aggregation. Science 284(5411):99–101

Parrish JK, Viscido SV, Grunbaum D (2002) Self-Organized Fish Schools: An Examination of Emergent Properties. Biological Bulletin 202(3):296–305

Partridge BL (1982) The Structure and Function of Fish Schools. Scientific American 246(6):114–123

Pitcher TJ, Partridge BL, Wardle CS (1976) A blind fish can school. Science 194(4268):963–965

Rands SA, Pettifor RA, Rowcliffe JM, Cowlishaw G (2004) State-dependent foraging rules for social animals in selfish herds. Proceedings of the Royal Society of London Series B: Biological Sciences 271(1557):2613–2620

Rauch EM, Millonas MM, Chialvo DR (1995) Pattern formation and functionality in swarm models. Physics Letters A 207(3–4):185–193

Reynolds CW (1987) Flocks, herds and schools: A distributed behavioral model. Computer Graphics 21(4):25–34

Reynolds CW (2004) Opensteer: Steering behaviors for autonomous characters. http://opensteer.sourceforge.net/, Last viewed in March 2012

Shimoyama N, Sugawara K, Mizuguchi T, Hayakawa Y, Sano M (1996) Collective Motion in a System of Motile Elements. Physical Review Letters 76(20):3870–3873

Simons AM (2004) Many wrongs: the advantage of group navigation. Trends in Ecology & Evolution 19(9):453–455

Smith J, Martin A (2009) Comparison of Hard-Core and Soft-Core Potentials for Modelling Flocking in Free Space. ArXiv e-prints

Sugawara K (2012) Personal communications

Szabó P, Nagy M, Vicsek T (2008) Turning with the others: novel transitions in an SPP model with coupling of accelerations. In: Proceedings of the IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO '08), Venice, Italy, pp 463–464

Szabó P, Nagy M, Vicsek T (2009) Transitions in a self-propelled-particles model with coupling of accelerations. Physical Review E 79(2):0219,080–021,913

Tanner HG, Jadbabaie A, Pappas GJ (2003a) Stable Flocking of Mobile Agents, Part I: Fixed Topology. In: Proceedings of the IEEE Conference on Decision and Control, pp 2010–2015

Tanner HG, Jadbabaie A, Pappas GJ (2003b) Stable Flocking of Mobile Agents, Part II: Dynamic Topology. In: Proceedings of the IEEE Conference on Decision and Control, pp 2016–2021

Toner J, Tu Y (1998) Flocks, herds, and schools: A quantitative theory of flocking. Physical Review E 58(4):4828–4858

Turgut A, Çelikkanat H, Gökçe F, Şahin E (2008) Self-Organized flocking in mobile robot swarms. Swarm Intelligence 2(2–4):97–120

Vaughan R, Sumpter N, Henderson J, Frost A, Cameron S (2000) Experiments in Automatic Flock Control. Robotics and Autonomous Systems 31:109–117

Vicsek T, Zafeiris A (2012) Collective motion. Physics Reports 517(34):71 – 140

Vicsek T, Czirók A, Ben-Jacob E, Cohen I, Shochet O (1995) Novel Type of Phase Transition in a System of Self-Driven Particles. Physical Review Letters 75(6):1226–1229

Vine I (1971) Risk of visual detection and pursuit by a predator and the selective advantage of flocking behaviour. Journal of Theoretical Biology 30(2):405–422

Viscido SV, Wethey DS (2002) Quantitative analysis of fiddler crab flock movement: evidence for 'selfish herd' behaviour. Animal Behaviour 63(4):735–741

Viscido SV, Miller M, Wethey DS (2002) The Dilemma of the Selfish Herd: The Search for a Realistic Movement Rule. Journal of Theoretical Biology 217(2):183–194

Warburton K, Lazarus J (1991) Tendency-distance models of social cohesion in animal groups. Journal of Theoretical Biology 150(4):473–488

Whitfield DP (2003) Redshank *Tringa totanus* flocking behaviour, distance from cover and vulnerability to sparrowhawk *Accipiter nisus* predation. Journal of Avian Biology 34(7):163–169

Wood AJ, Ackland GJ (2007) Evolving the selfish herd: emergence of distinct aggregating strategies in an individual-based model. Proceedings of the Royal Society of London Series B: Biological Sciences 274(1618):1637–1642