2020 International Symposium on Nonlinear Theory and Its Applications,
NOLTA2020, Virtual, November 16-19, 2020

Algorithm for Analyzing Swarm Formation in Two Dimensional Time-invariant External Potential

Yanran Wang† and Takashi Hikihara†

†Department of Electrical Engineering, Kyoto University Katsura, Nishikyo-ku, Kyoto, 615–8510 Japan
Email: y-wang@dove.kuee.kyoto-u.ac.jp, hikihara.takashi.2n@kyoto-u.ac.jp

Abstract—Clustering has the desirable properties for designing energy efficient Wireless Senor Networks protocols. In this paper, we investigate how cluster formation of sensors respond to external time-invariant potential. Mathematically, we link the sensors' trajectories to sectional curvature of the external potential manifold. Information of the external potential can be obtained by analyzing the formation change of Wireless Sensor Network clusters.

1. Introduction

In recent years, Wireless Sensor Networks (WSNs) have attracted much attention due to its ability to provide ubiquitous and multi-faceted situational awareness with a host of applications ranging from structural health monitoring, habitat surveillance, and target detection to power system management, smart car parking, and wireless luggage tags [1], [2], [6], [7], [10]. WSN depends on spatially distributed sensor node to measure and collect the desire environmental data within its sensing range, and then transmit to a control center called Base Station (BS). In most cases, energy is limited due to either hardward limitations or environments. As energy is constrained and data transmission is most energy costly, WSNs algorithms need to be architect in ways where data transmission, especially to BS, is minimized.

The existing algorithms focus on clustering of sensors, where collected data are first transmited to sensors with higher operational hierarchy, which then relay information to BS [4], [8]. To address the energy efficiency challenge from a new perspective, we focus on designing WSNs algorithm where desired environmental information is obtained by oberserving the change in sensor cluster formations. Quantitative environmental data transmission from WSNs to BS is eliminated in this algorithms, thereby minimizing the energy expenditure and extending network lifetime.

In this paper, we identify the envrionmental information as a potential hypersuface M of $(n-1)$-dimensions ($n = 3$). Mathematically, M is a Riemannian manifold defined by the set of solutions to a single equation

$$f(x_1, \ldots, x_n) = 0, \tag{1}$$

where f is a C^∞ function. We introduce an algorithm that exploits formation of the WSN sensors to compute curvature, which is invariant under isometry, of the manifold M. Curvature of manifold M can be seen as a manifestation of the environmental potential information.

2. Mathematical preliminaries

In this section, the theoretical basis for the algorithm are provided; necessary elements and definition of differential geometry are discussed [3], [5].

2.1. Metric, connection, and curvature

Let $U \subseteq \mathbb{R}^n$ be a non-empty open subset and $f : U \to \mathbb{R}$ a C^∞ function. Let $M \subseteq U \times \mathbb{R}$ be the graph of f. The closed subset M in $U \times \mathbb{R}$ projects homeomorphically onto U with inverse $(x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_n, f(x_1, \ldots, x_n))$ that is a smooth mapping from U to $U \times \mathbb{R}$. M is a closed smooth submanifold of $U \times \mathbb{R}$. Using the standard Riemannian metric on $U \times \mathbb{R} \subseteq \mathbb{R}^{n+1}$, the induced metric g on M at a point $p \in M$ is

$$g(p) = \langle \partial_{q_i}|_p, \partial_{q_j}|_p \rangle_p \, dq_i(p) \otimes dq_j(p) \tag{2}$$

with coordinate chart $\{q_i\}$ on M. Each $\partial_{q_i}|_p \in T_pM$ can be represented as a linear combination of $\{\partial_{x_i}|_p\} \in T_p(\mathbb{R}^{n+1})$, given as

$$\partial_{q_i}|_p = \partial_{x_i}|_p + \partial_{x_i} f(p) \partial_{x_{n+1}}|_p. \tag{3}$$

Consider the aforementioned graph M as a C^∞ Riemannian manifold. Given a curve, $C : [a, b] \longrightarrow M$, a *vector field* X along C is any section of TM over C ($X : [a, b] \longrightarrow TM$, projection $\pi : TM \longrightarrow M$, such that $\pi \circ X = C$). If M is a smooth manifold, all vector field on the manifold are also smooth. We denote the collection of all smooth vector fields on manifold M as $\mathfrak{X}(M)$.

For a Riemannian manifold (M, g), the *Levi-Civita connection* ∇_g on M is the unique connection on the tangent bundle TM that has both metric compatibility and torsion freeness. The Christoffel symbols of the second kind are the connection coefficients (in a

local chart) of the Levi-Civita connection denoted as

$$\Gamma^l_{jk} = \frac{1}{2} g^{lr}(\partial_k g_{rj} + \partial_j g_{rk} - \partial_r g_{jk}). \tag{4}$$

For a Riemannian manifold (M, g), a curve is called *geodesic* with respect to the connection ∇_g if its acceleration it zero. That is a curve γ where $\nabla_{\dot\gamma}\dot\gamma = 0$. A geodesic curve in n-dimensional Riemannian manifold can be expressed as a system of second order ordinary differential equations,

$$\frac{d^2\gamma^i}{dt^2} + \Gamma^i_{jk}\frac{d\gamma^j}{dt}\frac{d\gamma^k}{dt} = 0. \tag{5}$$

All geodesics are the shortest path between any two points on the manifold.

Any connection on a manifold gives rise to a *curvature tensor*. In some sense, curvature tensor measures how closely is the connection of the manifold relates to the standard connection in $\mathbb{R}^n$, which we assume has zero curvature. Let (M, g) be a Riemannian manifold and ∇ its Riemannian connection. The curvature tensor is a $(1, 3)$ tensor defined through the Lie bracket as

$$R(X, Y)Z = (\nabla_{[X,Y]} - [\nabla_X, \nabla_Y])Z, \tag{6}$$

where $X, Y, Z \in \mathfrak{X}(M)$, and $R(X, Y)Z$ is vector-valued. $R(X, Y)Z$ can be express in local chart as

$$R^k_{nij} = \frac{\partial\Gamma^k_{nj}}{\partial x^i} - \frac{\partial\Gamma^k_{ni}}{\partial x^j} + \Gamma^a_{nj}\Gamma^k_{ai} - \Gamma^a_{ni}\Gamma^k_{aj}. \tag{7}$$

Let tangent 2-plane, Π_p, be the two dimensional subspace in T_pM defined as $\Pi_p \equiv span\{u, v\}$, with $u, v \in T_pM$. *Sectional curvature* K of (M, g) at a point $p \in M$ with respect to the plane Π_p is defined as

$$K(\Pi_p) = \frac{\langle R(X, Y)X, Y\rangle_p}{|X|^2_p|Y|^2_p - \langle X, Y\rangle^2_p}, \tag{8}$$

where $X, Y \in \mathfrak{X}(M)$.

2.2. Jacobi field

Let $\gamma : [a, b] \longrightarrow M$ be geodesic. A vector field J along γ is called a Jacobi field if

$$\ddot J + R(J, \dot\gamma)\dot\gamma = 0, \tag{9}$$

where $\ddot J \equiv \nabla_{\frac{d}{dt}}\nabla_{\frac{d}{dt}}J$, and $\dot\gamma \equiv \frac{d\gamma}{dt}$.

2.3. Geodesic deviation equation

Geodesic deviation equation relates the acceleration of the separation vector between two neighbouring geodesic curves to Riemannian curvature tensor. Consider two nearby geodesics γ and $\widetilde\gamma$. In terms of local chart and curve parameter τ, the small coordinate difference can be written as vectors,

$$S(\tau) = \widetilde\gamma(\tau) - \gamma(\tau) \tag{10}$$

denote as seperation vector.

The separation acceleration is

$$W = \nabla_{\dot\gamma}\nabla_{\dot\gamma}S = \nabla_{\dot\gamma}V, \tag{11}$$

where V is the separation velocity. In a local chart, W and V can be expressed as

$$V^i = \frac{dS^i}{d\tau} + \Gamma^i_{jk}\dot\gamma^j S^k, \tag{12}$$

$$W^i = \frac{dV^i}{d\tau} + \Gamma^i_{lm}\dot\gamma^l V^m, \tag{13}$$

where $\dot\gamma \equiv \frac{d\gamma}{d\tau}$. Combining (12) and (13) gives

$$\begin{aligned} W^i = {} & \frac{d^2S^i}{d\tau^2} + 2\Gamma^i_{jk}\dot\gamma^j\frac{dS^k}{d\tau} + \frac{\partial\Gamma^i_{jk}}{\partial\gamma^l}\dot\gamma^l\dot\gamma^j S^k \\ & + \Gamma^i_{jk}\ddot\gamma^j S^k + \Gamma^i_{lm}\Gamma^m_{jk}\dot\gamma^j\dot\gamma^l S^k, \end{aligned} \tag{14}$$

where $\ddot\gamma \equiv \frac{d^2\gamma}{d\tau^2}$.

Rearranging (10) to be

$$\widetilde\gamma(\tau) = \gamma(\tau) + S(\tau), \tag{15}$$

where separation vector $S(\tau)$ is considered as a small variation parameter. Both curves γ and $\widetilde\gamma$ are geodesics, then by definition of (5),

$$\frac{d^2\gamma^i}{d\tau^2} + \Gamma^i_{jk}\frac{d\gamma^j}{d\tau}\frac{d\gamma^k}{d\tau} = 0, \tag{16}$$

and

$$\frac{d^2(\gamma^i + S^i)}{d\tau^2} + \Gamma^i_{jk}(\gamma^i + S^i)\frac{d(\gamma^j + S^j)}{d\tau}\frac{d(\gamma^k + S^k)}{d\tau} = 0. \tag{17}$$

Treating deviation vector $S^i(\tau)$ as the expansion parameter and only keep the first order terms (since it is small), and make use of (16), (17) becomes

$$0 = \frac{d^2S^i}{d\tau^2} + 2\Gamma^i_{jk}\dot\gamma^j\frac{dS^k}{d\tau} + \frac{\partial\Gamma^i_{jk}}{\partial\gamma^l}\dot\gamma^j\dot\gamma^k S^l. \tag{18}$$

Inserting both (16) and (18) into (14), the result is

$$\begin{aligned} W^i &= -(\frac{\partial\Gamma^i_{jl}}{\partial\gamma^k} - \frac{\partial\Gamma^i_{jk}}{\partial\gamma^l} + \Gamma^m_{jl}\Gamma^i_{mk} - \Gamma^i_{lm}\Gamma^m_{jk})\dot\gamma^j S^k\dot\gamma^l \\ &\equiv -R^i_{jkl}\dot\gamma^j S^k\dot\gamma^l. \end{aligned} \tag{19}$$

Note the deviation vector field S is a Jacobi field.

3. Swarm formation

For the definition of swarm formation we follow Olfati-Saber [9]. The topology of swarm formation is an example of undirected graph. Define a *graph* G as a pair $(\mathcal{V}, \mathcal{E})$ that consists of a set of vertices $\mathcal{V} = \{1, 2..., n\}$ and a set of edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$, and symmetric adjacency matrix A satisfying the property $a_{ij} \neq 0 \iff (i, j) \in \mathcal{E}$. Each vertices represent an agent, while the edges are the inter-agent distance. Let $q_i \in \mathbb{R}$ denote the position of agent i for all $i \in \mathcal{V}$. The vector $q = (q_1, ..., q_n)^T$ is the configuration of all agents of the swarm. To maintain identical inter-agent distances (geodesic length between two points on a manifold) over $G(q)$, consider an algebraic constraint,

$$dis(q_j - q_i) = d, \qquad \forall j \in N_i(q), \qquad d \in \mathbb{R}. \tag{20}$$

For simplicity, this paper focuses on line formations, where $N_i(q)$ is the ith agent most immediate left and right neighbouring agents. A configuration q that satisfying the set of constraints in (20) is refered as a *lattice formation.*

4. Algorithm

Equipped with the theoritical basis in Section 2, and the definition of cluster formation in Section 3, we will proceed to the algorithm itself. The algorithm uses the spatial-variant swarm's configuration in lattice formation to calculate the sectional curvature of the external energy potential. For simplicity, agents are divided into one head agent and the following agents. The head agent has a predefined initial position and velocity, and are not communicating with other agents. The following agent are assumed only to communicate with its immediate neighbours (which ever closer to the head agent). The distance constraint in Riemann manifold are the geodesic distance between two agent, which can also be seen as the inter-agent energy potential.

The algorithm has four main steps:

1. Define head agent trajectory.
2. Construct parallel vector field orthogonal to head agent trajectory.
3. Construct geodesic deviation vector field.
4. Calculate accelerations of the geodesic deviation vectors.

The mathematical details and WSNs interpretations are explained as the follows.

4.1. Head agent trajectory

Head agent trajectory, $h(t)$, on n-dimensional Riemannian manifold (M, g) is defind in coordinate chart as

$$\ddot{h}^k + \Gamma^k_{ij} \dot{h}^i \dot{h}^j = 0, \tag{21}$$

where $\dot{h}$ and $\ddot{h}$ are the first and second derivative with respect to time, respectively. For two dimensional manifold, the head agent trajectory is a system of ordinary differential equations

$$\begin{aligned} \dot{h}_1 &= h_3, \\ \dot{h}_2 &= h_4, \\ \dot{h}_3 &= -\Gamma^x_{xx}(h_3)^2 - 2\Gamma^x_{xy} h_3 h_4 - \Gamma^x_{yy}(h_4)^2, \\ \dot{h}_4 &= -\Gamma^y_{xx}(h_3)^2 - 2\Gamma^y_{xy} h_3 h_4 - \Gamma^y_{yy}(h_4)^2, \end{aligned} \tag{22}$$

where basis $\{x, y\}$ are used in the index. Note that the head agent trajectory is a geodesic in M.

4.2. Inter-agent energy potential

The agents are assumed to be in lattice formation. The distance constraint between two agents is measured to be the length of the geodesic. A parallel vector field, V, that is orthogonal to the head agent trajectory is constructed as

$$\nabla_V V = 0. \tag{23}$$

V forms a family of geodesic curves that is orthogonal to the head agent trajectory.

4.3. Following agent trajectories

As mentioned in section 2.3, S is the separation vector field. From the view of the WSN, following agents' trajectories are the integral curves of S. Different integral curves of S represent following agent's trajectory at different inter-agent potential energy level. Fig. 1 shows vector field S in two dimensions for an elliptic paraboloid potential manifold.

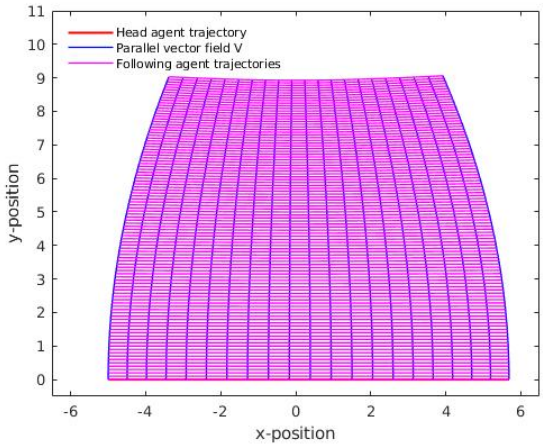


Figure 1: Two dimensional visual example of energy deviation lines respect to predefined head agent trajectory for elliptic paraboloid potential manifold.

4.4. Sectional curvature and acceleration vector field

The acceleration of the separation vectors along V, is equivalent to the change in the difference of nearby agent's velocity. According to (14), we can use this change to calculate the curvature of the manifold. Consider the sectional curvature equation (8) and substitude in the vector fields V and S constructed in section 4.2 and 4.3,

$$K(V,S) = \frac{\langle R(V,S)V,S\rangle}{|V|^2|S|^2 - \langle V,S\rangle^2}. \tag{24}$$

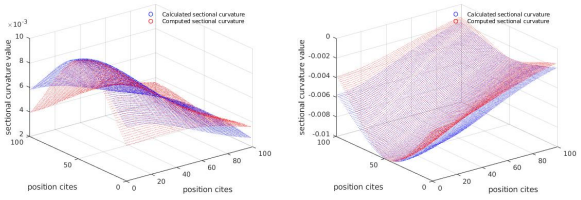
Also S is in fact a Jacobi field along V that is also orthogonal to V. That is $W \equiv \nabla_\tau^2 S = -R(V,S)V$, and $\langle V,S\rangle = 0$. Combining with the fact that the integral curves of V are geodesics set to have velocity $|\dot\gamma| = 1$, the equation for sectional curvature is simplified to

$$W = -K(V,S)S. \tag{25}$$

For a two dimensional Riemannian manifold (M,g), there is only one sectional curvature at each point $p \in M$.

5. Simulations

Simulations for the algorithm are done on various two dimensional potential manifold. Overall, the algorithm is able to compute sectional curvature with some extend of accuracy, shown in Fig. 2.



(a) $F(x_1,x_2) = \dfrac{x_1^2 + x_2^2}{a}$. (b) $F(x_1,x_2) = \dfrac{x_1^2 - x_2^2}{a}$.

Figure 2: Sectional curvatures comparison with algorithm computed (red) and metrically calculated (blue) in 3D for (a) elliptic paraboloid and (b) hyperbolic paraboloid potential manifold. Head agent initial condition $h = [-5,0], \dot h = [1,0]$, and parameter $a = 20$.

6. Conclusion

This paper demostrates a new approach for energy efficient WSNs' algorithm. Unlike traditional algorithms, the proposed algorithm relies on cluster formation of WSNs to obtain environmental information, thereby eliminates the neccessity of Base Station data transmissions. However, the ability to observe the change of WSNs cluster formation is requested. By using the change of swarm lattic formation through an external potential energy, the algorithm is able to estimate the sectional curvature of the external potential.

References

[1] M. Erol-Kantarci and H. T. Mouftah. Wireless sensor networks for cost-efficient residential energy management in the smart grid. *IEEE Transactions on Smart Grid*, 2(2):314–325, 2011.

[2] Y. Gao and R. Jin. A novel wireless sensor networks platform for habitat surveillance. In *CSSE: International conference on Computer Science and Software Engineering*, volume 4, pages 1028–1031. IEEE.

[3] V. G. Ivancevic and T. T. Ivancevic. *Applied Differential Geometry: A Modern Introduction*. World Scientific Publishing Co. Pte. Ltd., Singapore, 2007.

[4] Y. Jin, L. Wang, Y. Kim, and X. Yang. Eemc: An energy-efficient multi-level clustering algorithm for large-scale wireless sensor networks. *Computer Networks*, 52(3):542–562, 2008.

[5] J. M. Lee. *Riemannian Manifolds: An Introduction to Curvature*. Springer-Verlag New York, Inc., New York, USA, 1997.

[6] D. Li, K. Wong, Y. H. Hu, and A. Sayeed. Detection, classification, and tracking of targets. *IEEE Signal Processing Magazine*, 19(2):17–29, 2002.

[7] Q. Ling, Z. Tian, Y. Yin, and Y. Li. Localized structural health monitoring using energy-efficient wireless sensor networks. *IEEE Sensors Journal*, 9(11):1596–1604, 2009.

[8] M. Mehdi Afsar and M.-H. Tayarani-N. Clustering in sensor networks: A literature survey. *Journal of Network and Computer Applications*, 46:198–226, 2014.

[9] R. Olfati-Saber. Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, 2006.

[10] S. Srikanth, P. Pramod, K. Dileep, S. Tapas, M. U. Patil, and C. B. N. Sarat. Design and implementation of a prototype smart parking (spark) system using wireless sensor networks. In *2009 International Conference on Advanced Information Networking and Applications Workshops*, pages 401–406. IEEE.