A Class of Distributed Online Aggregative Optimization in Unknown Dynamic Environment

Chengqian Yang 1, Shuang Wang 1, Shuang Zhang 2, Shiwei Lin 2,*, and Bomin Huang 2,*

1 School of Control Engineering, Northeastern University at Qinhuangdao, Qinhuangdao 066004, China
2 College of Computer Engineering, Jimei University, Xiamen 361021, China
* Correspondence: huangbomin01@jmu.edu.cn

Abstract: This paper considers a class of distributed online aggregative optimization problems over an undirected and connected network. It takes into account an unknown dynamic environment and some aggregation functions, which is different from the problem formulation of the existing approach, making the aggregative optimization problem more challenging. A distributed online optimization algorithm is designed for the considered problem via the mirror descent algorithm and the distributed average tracking method. In particular, the dynamic environment and the gradient are estimated by the averaged tracking methods, and then an online optimization algorithm is designed via a dynamic mirror descent method. It is shown that the dynamic regret is bounded in the order of $O(\sqrt{T})$. Finally, the effectiveness of the designed algorithm is verified by some simulations of cooperative control of a multi-robot system.

Keywords: online optimization; aggregated terms; distributed algorithm; dynamic environment

MSC: 68W15

1. Introduction

The distributed online optimization problem for multi-agent systems has received considerable attention in the past few decades [1,2]. The objective is to minimize a global time-varying cost function, written as the sum of local convex functions, under the constraint that each individual agent has only knowledge of the local convex functions.

Recently, some researchers have focused on distributed online aggregative optimization, which is a special class of online optimization problems. In distributed online aggregative optimization, the local cost functions include some aggregative terms motivated by many real applications. These aggregative terms make the design and analyses of the online optimization algorithm more challenging. Some tracking methods were presented to solve distributed online aggregative optimization problems. For example, a distributed aggregative gradient tracking algorithm is proposed and analyzed in [3] to solve a distributed online aggregative optimization problem. It is shown that convergence to the optimal variable is linear. Different from [3], a set constraint is considered in [4], where an online distributed gradient tracking algorithm is proposed to solve a distributed online aggregative optimization problem with exact gradient information and stochastic/noisy gradients, respectively. The upper boundary of dynamic regret is analyzed and given, and some simulations on the target surrounding problem are provided to verify the effectiveness of the designed algorithm. The authors of [5] considered the distributed online aggregative optimization without the assumption of boundedness of the gradients and the feasible sets. In particular, a projected aggregative tracking algorithm was presented in [5], and the authors showed that the dynamic regret is bounded by a constant term and a term related to time variations.

Dynamic environments are common in the real applications of online optimization [6–11]. For example, the authors of [6] showed that many online learning problems, including the
dynamic texture analysis, solar flare detection, sequential compressed sensing of a dynamic scene, traffic surveillance, tracking self-exciting point processes, and network behavior in the Enron email corpus, can benefit from the incorporation of a dynamic environment. In [7], the tracking problem of a time-varying parameter with unknown dynamics was studied as an online optimization in a dynamic environment. The distributed tracking and tracking of dynamic point processes on network problems were solved in [8,9] using mirror descent methods with dynamical environments, respectively. The localization of sensor networks problem was solved in [10] by using a distributed online bandit learning algorithm over a multi-agent network with a dynamical environment. Note that the dynamic environments were assumed to be known in [6–10]. In contrast with [6–10], an unknown dynamic environment was considered in [11], where the dynamic regret was shown to be bounded.

Based on the above discussions, the distributed online aggregative optimization with a dynamic environment is motivated by many real applications. The problem has many applications, especially under the condition that the dynamic environment is unknown. Therefore, this paper considers a class of distributed online aggregative optimization problems with an unknown dynamic environment. The main contributions are as follows. Some aggregative terms and an unknown dynamic environment are simultaneously considered in this paper for an online optimization problem. The problem formulation comes from some real applications, e.g., the cooperative control problem for a multi-robot system in the simulation section. Compared with [3–11], the dynamic environment was not considered in [3–5], and the aggregative terms were not considered in [6–11], respectively. Therefore, the design and analyses of the optimization algorithm in this paper are more challenging than in [3–11]. In particular, some averaged tracking methods estimate the dynamic environment and the gradient. Based on the estimations, an online optimization algorithm is designed via a dynamic mirror descent method. It is shown that the dynamic regret is bounded in the order of $O(\sqrt{T})$, and some simulations are proposed to verify the effectiveness of the designed algorithm.

The remaining parts of this paper are organized as follows. In Sections 2 and 3, some preliminaries and problem formulation are presented. The main results are proposed in Section 4. In Section 5, the performance of the designed algorithm is verified by some simulations. Section 6 concludes the paper.

2. Preliminaries

2.1. Notations

$\mathbb{R}^n$ denotes the $n$-dimensional Euclidean space. $||x||$ denotes the 2-norm of a vector $x$. $[T]$ with positive integer $T$ denotes the set $\{1,2,\ldots,T\}$. $\langle x,y \rangle$ denotes the inner product of vectors $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^n$, i.e., $\langle x,y \rangle = x^T y$ where $x^T$ denotes the transpose of $x$. For a function $f(x,t) : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}$, $\nabla f(x,t)$ denotes the gradient of $f(x,t)$ with respect to a vector $x$. $\sigma_2(W)$ denotes the second-largest singular value of matrix $W$.

2.2. Graph Theory

For a multi-agent system, we use a directed graph $G = (\mathcal{V}, \mathcal{E})$ to describe the information exchange within it, where $\mathcal{V} = \{1,2,\ldots,N\}$ is the node set and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the edge set. $(j,i) \in \mathcal{E}$ represents that agent $j$ can obtain information from agent $i$. The self-loop is unconsidered, i.e., $(i,i) \notin \mathcal{E}$. $\mathcal{N}_i = \{ j : (j,i) \in \mathcal{E} \}$ denotes the in-neighbor set of node $i$. If for every $(j,i) \in \mathcal{E}$, there exists $(i,j) \in \mathcal{E}$, then the graph is an undirected graph. A path between nodes $i$ and $j$ is a sequence of edges $(i,\mathcal{V}_1), (\mathcal{V}_1, \mathcal{V}_2), \ldots, (\mathcal{V}_k,j)$ in the graph $G$ where $\mathcal{V}_l, l = 1,2,\ldots,k$ are some distinct nodes. An undirected graph is connected if there is a path between each pair of nodes.
2.3. Bregman Divergence

The mirror descent algorithm based on Bregman divergence is frequently used and effective in online optimization [8]. The Bregman divergence is defined as follows. Let \( R : \mathbb{R}^n \to \mathbb{R} \) be a strongly convex function which satisfies
\[
R(x) \geq R(y) + \langle x - y, \nabla R(y) \rangle + \frac{1}{2\alpha} \|x - y\|^2, \quad \forall x, y \in \mathbb{R}^n, \tag{1}
\]
and define the Bregman divergence \( D_R(x, y) \) as
\[
D_R(x, y) = R(x) - R(y) - \langle x - y, \nabla R(y) \rangle. \tag{2}
\]

3. Problem Formulation

Considering a network system composed of \( N \) agents, and there exists a sequence of time-varying convex function \( f_i(x) : \mathbb{R}^n \to \mathbb{R} \) over the network composed of \( f_{i,t}(x, v(x)) : \mathbb{R}^n \times \mathbb{R}^d \to \mathbb{R} \). Considering the following optimization problem,
\[
\min_{x \in \mathcal{X}} f_i(x) = \sum_{i=1}^{N} f_{i,t}(x, v(x)), \tag{3}
\]
where \( i \in [N], t \in [T], x \in \mathcal{X} \subseteq \mathbb{R}^n, \psi : \mathbb{R}^n \to \mathbb{R}^d, \) and \( v : \mathbb{R}^n \to \mathbb{R}^d \) is the aggregative variable defined by \( v(x) = \frac{1}{N} \sum_{i=1}^{N} \psi_i(x) \). The function \( f_{i,t}(x, v(x)) \) is convex and assigned to agent \( i \). Supposing there exists a sequence of unknown stable non-expansive mapping \( A_i \in \mathbb{R}^{n \times n} \), i.e., \( \|A_i\| \leq 1 \), such that
\[
x^*_i = A_i x^*_i + \theta_i, \tag{4}
\]
where \( x^*_i = \arg\min_{x \in \mathcal{X}} f_i(x) \) and \( \theta_i \) is an unstructured and unknown noise. Assuming that agent \( i \) independently observes the mapping \( A_i \), and \( A_{i,t} \) denotes the observed value in time \( t \). Moreover, assuming that \( A^*_i \) is the optimal observed value for all agents and \( A^*_i \) is the optimal solution to the following optimization problem:
\[
A_i^* = \arg\min_{A_i} \sum_{i=1}^{N} \|A_i - A_{i,t}\|^2. \tag{5}
\]

It follows from (4) and (5) that
\[
x^*_{i+1} = A_i x^*_i + \theta_i, \tag{6}
\]
where \( \theta_i = (A_i - A^*_i) x^*_i + \theta_i \).

**Distributed Online Aggregative Optimization Problem:**

The objective is to generate a sequence \( x_{i,t}, i \in [N], t \in [T] \) to minimize the following dynamic regret,
\[
\text{Reg}(T) := \sum_{t=1}^{T} \sum_{i=1}^{N} f_{i,t}(x_{i,t}) - \sum_{t=1}^{T} \sum_{i=1}^{N} f_{i,t}(x^*_i), \tag{7}
\]
where \( x^*_i \) is the optimal solution to (3) which satisfies constraint (6), i.e., \( x^*_i = \arg\min_{x \in \mathcal{X}} f_i(x) \) while \( x^*_i \) satisfies constraint (6).

**Remark 1.** The online aggregative optimization problems have been studied in [4,5]. However, this paper considers an unknown dynamic environment, which is more challenging than [4,5], i.e., the mapping \( A_i \).

Let \( \nabla \psi_i(x_i), \nabla_1 f_{i,t}(x_i, v) \) and \( \nabla_2 f_{i,t}(x_i, v) \) denote \( \nabla_x \psi_i(x_i), \nabla_x f_{i,t}(x_i, v) \) and \( \nabla_v f_{i,t}(x_i, v) \), respectively. Let \( W \in \mathbb{R}^{N \times N} \) be the adjacency matrix of network \( G \), and
\( \eta_t \in \mathbb{R}, \ t \in [T] \) be the global step-size which will be used to design an online optimization algorithm in the next section. Let \( \mathcal{X} \subseteq \mathbb{R}^n \) and \( \mathcal{Y} \subseteq \mathbb{R}^d \) be some convex sets which will be defined in the next section. Some necessary assumptions are as follows.

**Assumption 1.** Graph \( \mathcal{G} \) is undirected and connected, and \( W \) is doubly stochastic, i.e.,

\[
\sum_{i=1}^{N} W_{ij} = \sum_{j=1}^{N} W_{ij} = 1,
\]

and there exists a constant \( \alpha \in (0, 1) \) such that \( W_{ij} \geq \alpha \) when \( W_{ij} \geq 0 \), and \( W_{ii} \geq \alpha \) for all \( i \in [N] \).

**Assumption 2.** For \( x, m \in \mathcal{X}, y, n \in \mathcal{Y}, i \in [N] \) and \( t \in [T] \), the functions \( f_{it}(x) \), \( \nabla_1 f_{it}(x, y) \), \( \nabla_2 f_{it}(x, y) \), \( \psi_i(x) \), and \( \nabla \psi_i(x) \) are Lipschitz continuous, and the functions \( \psi_i(x) \), \( \nabla_1 f_{it}(x, y) \) and \( \nabla_2 f_{it}(x, y) \) are bounded, i.e.,

\[
\begin{align*}
\| x - m \| & \leq \| f_{it}(x) - f_{it}(m) \| \leq L \| x - m \| \\
\| \nabla_1 f_{it}(x, y) - \nabla_1 f_{it}(m, n) \| & \leq L(\| x - m \| + \| y - n \|) \\
\| \nabla_2 f_{it}(x, y) - \nabla_2 f_{it}(m, n) \| & \leq L(\| x - m \| + \| y - n \|) \\
\| \psi_i(x) - \psi_i(m) \| & \leq L \| x - m \| \\
\| \nabla \psi_i(x) - \nabla \psi_i(m) \| & \leq L \| x - m \| \\
\| \psi_i(x) \| & \leq H, \| \nabla_1 f_{it}(x, y) \| \leq H, \| \nabla_2 f_{it}(x, y) \| \leq H,
\end{align*}
\]

where \( L \) is the Lipschitz constant, \( \iota > 0 \) and \( H > 0 \).

**Assumption 3.** There exists

\[
| D_R(x, z) - D_R(y, z) | \leq K \| x - y \| \\
D_R(x, \sum_{i=1}^{N} a(i)y_i) \leq \sum_{i=1}^{N} a(i)D_R(x, y_i) \\
| \nabla R(x) - \nabla R(y) | \leq L_R \| x - y \|,
\]

where \( x, y, y_i, z \in \mathcal{X}, a(i) \) is on the \( N \)-dimensional simplex, and \( K \) and \( L_R \) are some Lipschitz constants.

**Assumption 4.** There exists positive constants \( M_A \) and \( M \), such that \( \| A_t^\ast - I_n \| \leq M_A \eta_t \), and \( \sum_{t=1}^{T} \| A_{t, t-1} \| \leq M \eta_t \), where \( \eta_t \) is the global step sequence. Moreover, \( D_R(A_t^\ast x, A_t^\ast y) \leq D_R(x, y) \) holds for \( x, y \in \mathcal{X} \) and \( t \in [T] \).

**Remark 2.** Assumptions 1, 3, and 4 come from [6–8,11]. They are common in mirror descent algorithms for distributed online optimization. Assumption 2 is also common in online aggregation optimization [4].

### 4. Main Result

Inspired by [11], consider the following online algorithm:
This makes the design and analyses of the optimization algorithm more challenging. In particular, the dynamic regret of algorithm (8) will be presented as follows.

\[ A_{i,t} = \sum_{j=1}^{N} W_{ij} \dot{A}_{j,t-1} + A_{i,t} - A_{i,t-1} \]  
\[ x_{i,t+1} = \arg\min_{x \in \mathcal{X}} \{ \eta_t (x, \nabla x_{i,t}) + D_R(x, y_{i,t}) \} \]  
\[ x_{i,t+1} = \sum_{j=1}^{N} W_{ij} \dot{A}_{j,t} \]  
\[ y_{i,t+1} = \sum_{j=1}^{N} W_{ij} x_{j,t+1} \]  
\[ v_{i,t+1} = \sum_{j=1}^{N} W_{ij} v_{j,t} + \psi_t (x_{i,t+1}) - \psi_t (x_{i,t}) \]  
\[ z_{i,t+1} = \sum_{j=1}^{N} W_{ij} z_{j,t} + \nabla \psi_t (x_{i,t+1}) - \nabla \psi_t (x_{i,t}) \]

where \( i \in [N], t \in [T] \), \( \nabla x_{i,t} = \nabla f_{i,t}(x_{i,t}, v_{i,t}) + \nabla^2 f_{i,t}(x_{i,t}, v_{i,t}) z_{i,t} \). \( \eta_t \) is the global step sequence, \( \dot{x}_{i,t+1}, y_{i,t+1}, \dot{A}_{i,t+1} \) and \( v_{i,t+1} \) are some auxiliary variables, and \( x_{i,t+1} \) is the designed sequence. In (8), the initial states are chosen such that \( \dot{A}_{i,0} = A_{i,0} = 0 \), \( y_{i,0} = x_{i,0} = \dot{x}_{i,0} = 0 \), \( v_{i,0} = \psi_t (x_{i,0}) \) and \( z_{i,0} = \nabla \psi_t (x_{i,0}) \).

It follows from (8) that the states in (8) are bounded. Let sets \( \mathcal{X} \) and \( \mathcal{Y} \) satisfy \( \mathcal{X} \subseteq \mathcal{Y}, x_{i,t} \in \mathcal{X}, y_{i,t} \in \mathcal{Y} \) and \( v_{i,t} \in \mathcal{Y} \). Let \( P = \max_{i} \{ v_{i,t}, z_{i,t} \}, Q = \sup_{x \in \mathcal{X}} \| x \|, R^2 = \sup_{x, y \in \mathcal{X}} D_R(x, y) \), and

\[ \Delta A_{i,t} = \sum_{j=1}^{N} W_{ij} \dot{A}_{j,t} - A_{i,t} \]  
\[ x_{t+1} = \frac{1}{N} \sum_{i=1}^{N} x_{i,t+1}, \]  
\[ \nabla v(x) = \frac{1}{N} \sum_{i=1}^{N} \nabla \psi_t (x) \]  
\[ \nabla v(x) = \nabla f_{i,t}(x_{i,t}, v(x_{i,t})) + \nabla^2 f_{i,t}(x_{i,t}, v(x_{i,t})) \nabla v(x_{i,t}) \]  
\[ \nabla v(x) = \nabla f_{i,t}(x_{i,t}, v(x_{i,t})) + \nabla^2 f_{i,t}(x_{i,t}, v(x_{i,t})) z_{i,t} \]

Remark 3. We have provided a comparative analysis of problem formulation of online optimization in Table 1, where DAT denotes distributed average tracking. It follows that the problem considered in this paper is more challenging than [3–11]. Algorithms (8a)–(8d) come from [11]. In fact, algorithms (8a)–(8d) have been used in [11] to solve a class-distributed online optimization problem with an unknown mapping. However, the aggregative variable \( v(x) \) is not considered in [11]. This makes the design and analyses of the optimization algorithm more challenging. In particular, it follows from (3) that, for agent \( i \), the gradient of the function \( f_{i,t}(x, v(x)) \) with respect to \( x \) is \( \nabla f_{i,t}(x_{i,t}, v(x_{i,t})) + \nabla^2 f_{i,t}(x_{i,t}, v(x_{i,t})) \nabla v(x_{i,t}) \). But \( v(x_{i,t}) \) and \( \nabla v(x_{i,t}) \) are unknown for agent \( i \) due to the topology constraint. Therefore, the algorithms (8e) and (8f) are added by using the distributed average tracking method to estimate \( v(x_{i,t}) \) and \( \nabla v(x_{i,t}) \). Based on (8e) and (8f), the new gradient \( \nabla v_{i,t} = \nabla f_{i,t}(x_{i,t}, v_{i,t}) + \nabla^2 f_{i,t}(x_{i,t}, v_{i,t}) z_{i,t} \) is introduced in algorithm (8g). Hence, algorithms (8e) and (8f) are used to achieve aggregate gradient tracking via DAT. A strict analysis of dynamic regret of algorithm (8) will be presented as follows.
Table 1. Comparison of problem formulation and algorithm of online optimization.

<table>
<thead>
<tr>
<th>Reference</th>
<th>Aggregative Terms</th>
<th>Dynamic Environment</th>
<th>Algorithm/Technique</th>
</tr>
</thead>
<tbody>
<tr>
<td>[3]</td>
<td>Yes</td>
<td>No</td>
<td>Distributed aggregative gradient tracking</td>
</tr>
<tr>
<td>[4]</td>
<td>Yes</td>
<td>No</td>
<td>Online distributed gradient tracking</td>
</tr>
<tr>
<td>[5]</td>
<td>Yes</td>
<td>No</td>
<td>Projected aggregative tracking</td>
</tr>
<tr>
<td>[6]</td>
<td>No</td>
<td>Known</td>
<td>Dynamic mirror descent</td>
</tr>
<tr>
<td>[7]</td>
<td>No</td>
<td>Known</td>
<td>Online gradient descent</td>
</tr>
<tr>
<td>[8]</td>
<td>No</td>
<td>Known</td>
<td>Decentralized mirror descent</td>
</tr>
<tr>
<td>[10]</td>
<td>No</td>
<td>Known</td>
<td>Adaptive gradient method</td>
</tr>
<tr>
<td>[9]</td>
<td>No</td>
<td>Known</td>
<td>Distributed bandit online leaning</td>
</tr>
<tr>
<td>[11]</td>
<td>No</td>
<td>Unknown</td>
<td>Gradient tracking via DAT</td>
</tr>
<tr>
<td>This paper</td>
<td>Yes</td>
<td>Unknown</td>
<td>Aggregative gradient tracking via DAT</td>
</tr>
</tbody>
</table>

Some lemmas and the main result are presented as follows, and the proofs are given in the Appendixes A–G.

**Lemma 1.** Consider the mapping (5) and the algorithm (8a), and supposing that Assumptions 1 and 4 hold. Then for all \( i \in [N] \) and \( t \in [T] \), there exists

\[
\| \Delta A_{i,t} \| \leq \frac{8M\eta t}{1 - \lambda},
\]

where \( \lambda \in (0, 1) \).

**Lemma 2.** Considering algorithm (8f), and supposing that Assumptions 1 and 2 hold, then for all \( i \in [N] \) and \( t \in [T] \), there exists,

\[
\| z_{i,t} - z_t \|, \| \nabla v(x_{i,t}) \| \leq \Delta_1,
\]

where \( \Delta_1 = NP\gamma + \frac{2NL\beta}{1 - p} + 4L, \Delta_2 = H\Delta_1 + HL + H, \gamma = (1 - \frac{n}{2N})^{-2}, \) and \( \beta = 1 - \frac{n}{2N} \).

**Lemma 3.** Considering algorithms (8a)–(8d), and supposing that Assumptions 1–4 hold, then for all \( i \in [N] \) and \( t \in [T] \), there exists,

\[
\frac{x_{i,t+1} - \bar{x}_{t+1}}{\| x_{i,t+1} - x_{i,t} \|} \leq \Delta_3 \sum_{\tau=0}^{t} \sigma_2^{\tau-\tau}(W)\eta_\tau,
\]

where \( \Delta_3 = \sqrt{N} \left[ \Delta_2 + M_A Q + \frac{8MQ}{1 - \lambda} \right] \).

**Lemma 4.** Considering algorithm (8), and supposing that Assumptions 1–3 hold, then for all \( i \in [N] \) and \( t \in [T] \), there exists,

\[
\frac{x_{i,t+1} - x_{i,t}}{\| x_{i,t+1} - x_{i,t} \|} \leq \Delta_4 \eta_t + 2\Delta_3 \sum_{\tau=0}^{t} \sigma_2^{\tau-\tau}(W)\eta_\tau
\]

\[
\frac{v_{i,t} - v(x_{i,t})}{\| v_{i,t} - v(x_{i,t}) \|} \leq N\gamma P\beta^t + \Delta_5 \eta_{t-1} + \Delta_6 \sum_{\tau=0}^{t} \sigma_2^{\tau-\tau}(W)\eta_\tau
\]

\[
\frac{z_{i,t} - \nabla v(x_{i,t})}{\| z_{i,t} - \nabla v(x_{i,t}) \|} \leq N\gamma P\beta^t + \Delta_5 \eta_{t-1} + \Delta_6 \sum_{\tau=0}^{t} \sigma_2^{\tau-\tau}(W)\eta_\tau,
\]

where \( \Delta_4 = \Delta_2 + M_A Q + \frac{8MQ}{1 - \lambda}, \Delta_5 = L\Delta_4 \left( \frac{N\gamma \beta}{1 - p} + 2 \right), \) and \( \Delta_6 = 2L\Delta_3 \left( \frac{N\gamma \beta}{1 - p} + 3 \right). \)
Lemma 5. Considering algorithm (8), and supposing that Assumptions 1, 3 and 4 hold, then for all $i \in [N]$ and $t \in [T]$, there exists,

$$
\Delta T = \frac{8dMQ}{(1-\lambda)}.
$$

Lemma 6. Considering algorithm (8), and supposing that Assumptions 1–3 hold, then for all $i \in [N]$ and $t \in [T]$, there exists,

$$
\|\nabla_{i,t} - \nabla_{i,t}\| \leq (L + L^2 + H)\{N\gamma P\beta^t + \Delta \gamma_{i-1} + \Delta_0 \sum_{t=0}^t \mathcal{Q}^{-T}(W)\eta_t\}.
$$

The main result of this paper is as follows.

Theorem 1. Considering the optimization problem (3), constraint (6), and algorithm (8), and supposing that Assumptions 1–4 hold. Let $\eta_t = \eta > 0$ and $t > \Delta_T$. Then there exists

$$
\text{Reg}(T) \leq \frac{L}{t - \Delta_T} \left\{ \Delta_8 \eta T + \frac{2NR^2 + NKC_T}{\eta} + \Delta_9 \right\},
$$

where $C_T = \sum_{t=1}^T \|\theta_t\|$, $\Delta_8 = 2Q(L + L^2 + H)\{\Delta_5 + \frac{\Delta_6}{1 - \sigma_t(W)}\} + \frac{N(H + HL)^2}{2} + \frac{2N(H + HL)\Delta_3}{1 - \sigma_t(W)}$, and $\Delta_9 = 2NQ\Delta_7 + \Delta_7 C_T + 2QN^2\gamma P(L + L^2 + H)\frac{\beta}{\overline{\gamma}_R}$.

Remark 4. The condition $i > \Delta_T$ is reasonable since a sufficiently small $\Delta_T$ is chosen by decreasing $L_R$. It follows from the boundary $\text{Reg}(T)$ in Theorem 1 that a proper step is $\eta_t = \eta_0/\sqrt{T}$ with $\eta_0 > 0$. In such a case, the regret $\text{Reg}(T)$ is bounded in the order of $O(\sqrt{T})$ for any arbitrary $T > 1$.

5. Simulations

Inspired by [5,8], considering a cooperative control problem for a multi-robot system in a plant $\mathcal{X} \subseteq \mathbb{R}^2$, in particular, a moving target with state $T \in \mathcal{X}$ is modeled as

$$
T_{t+1} = A_T T_t + \theta_{i,t}^v,
$$

where $A_T \in \mathbb{R}^{2 \times 2}$ and $\theta_{i,t}^v \in \mathbb{R}^2$ is the noise. Moreover, there is a multi-robot network composed of $N$ robots aiming to protect the target, while there are $N$ intruders aiming to capture the target. Let $\mathcal{I}_i \in \mathcal{X}$ denote the state of the $i$th intruder at time $t$, which is assigned to robot $i$. Supposing that the formation of a multi-robot network is defined by a virtual leader–follower formation method, let $x_{i,t}^v$ denote the leader’s state which is unknown for all robots. In particular, let $x_{i,t} \in \mathcal{X}$ denote the estimated value of $x_{i,t}^v$ by robot $i$ at time $t$, and the state of robot $i$ is given by $\psi_i(x_{i,t}) = \beta_{i} x_{i,t} + \gamma_{i}$, where $\beta_{i}$ and $\gamma_{i}$ represent an offset of robot $i$ to the virtual leader $x_{i,t}^v$.

The multi-robot network is used to protect the moving target by choosing an optimal virtual leader such that the average state of all the robots is close to the target. Meanwhile, each robot is simultaneously close to the associated intruder, referring to Figure 1. Let $v_i(x_t) = \frac{1}{N} \sum_{i=1}^N \psi_i(x_{i,t})$ denote the averaged state of robots where $x_t = \text{col}(x_{1,t}, x_{2,t}, \ldots, x_{N,t})$. Then, the cooperative control problem can be modeled as the following optimization problem:

$$
\min_{x_{i,t} \in \mathcal{X}} f_i(x_t) = \sum_{i=1}^N f_i(x_{i,t}, v(x_t)),
$$

(13)
where
\[
f_{i,t}(x_i, v_t) = \alpha_1 \|v(x_i) - T_t\|^2 + \alpha_2 \|\psi_i(x_i, t) - I_i, t\|^2,
\]
\(\alpha_1 > 0\), and \(\alpha_2 > 0\). In fact, the objective is to cooperate in seeking an optimal virtual leader, which is the optimal solution to the problem (13).

\[f_t(x, v_t) = \alpha_1 \|v(x_i) - T_t\|^2 + \alpha_2 \|\psi_i(x_i, t) - I_i, t\|^2,\]
\(\alpha_1 > 0\), and \(\alpha_2 > 0\). In fact, the objective is to cooperate in seeking an optimal virtual leader, which is the optimal solution to the problem (13).

**Figure 1.** The concept of the cooperative control problem for a multi-robot system.

It is worth pointing out that the estimated virtual leader should be consensus. Therefore, let \(x^*_t \in \mathcal{X}\) be the consensus solution to problem (13). Then \(x^*_t\) should be the solution to the following online optimization problem:

\[x^*_t = \min_{x \in \mathcal{X}} f_t(x) = \sum_{i=1}^{N} f_{i,t}(x, v(x)),\] (14)

where \(\psi_i(x) = \beta_i x + \gamma_i\) and \(v(x) = \frac{1}{N} \sum_{i=1}^{N} \psi_i(x)\). In addition, it follows from (12) that \(x^*_t\) should also have the mapping \(A_t\), i.e., \(x^*_t\) satisfies the following constraint:

\[x^*_{t+1} = A_t x^*_t + \theta_t,\] (15)

where \(\theta_t \in \mathbb{R}^2\) is an unknown and unstructured noise. Note that \(A_t\) is unknown for all robots. Hence, \(A_{i,t}\) denotes the observed mapping of \(A_t\) at time \(t\) for robot \(i\). Problem (14) and constraint (15) take the form of problem (3) and constraint (6), so Algorithm 1 can be used to solve problem (14) with constraint (15).

Let \(N = 4\), \(\mathcal{X} = \{x \in \mathbb{R}^2 \|x\| \leq 3\}\), \(A_t = \begin{pmatrix} 1 & (1 + \cos t) \times 10^{-3} \\ 0 & 1 \end{pmatrix}\), \(\theta_t\) is a random vector chosen from a normal distribution with zero mean and standard deviation \(1; 2 \times 10^{-3}\), \(\mathcal{I}_{i,t} = T_i + \text{col}(\cos(2i\pi/N), \sin(2i\pi/N))\), \(\beta_i = 1\), \(\gamma_i = \text{col}(\cos(2i\pi/N), \sin(2i\pi/N))\), \(\alpha_1 = 0.1\), \(\alpha_2 = 0.2\), the unknown noise \(\theta_t\) satisfies \(\|\theta_t\| \leq 0.2\), and \(A_{i,t} = A_t + \mu_i I_2\) where \(\mu_i\) is a random number chosen from a normal distribution with zero mean and standard deviation \(10^{-3}\).

Choosing \(T = 1000\), \(\eta_t = 0.001\) and \(R(x) = 0.01 \|x\|^2\), and supposing that an arbitrary doubly stochastic \(W\) is given, it can be verified that Assumptions 1–4 and the condition \(\iota > \Delta_T\) hold. The simulations are performed on a computer equipped with AMD Ryzen 9 5950X 16-core CPU, 64G RAM, and Nvidia RTX 3080Ti GPU. Figure 2 shows the trajectory \(x_{1,t}\) of robot 1 by using Algorithm 1. The dynamic regret of Algorithm 1 is indicated in Figure 3, and the regret \(\text{Reg}(T)\) in Figure 3 is bounded. Therefore, Theorem 1 is verified by Figures 2 and 3.
2.2 2.4 2.6 2.8 3 3.2
Figure 2. Trajectories of $x^*_t$ and $x_{1,t}$ over $T = 1000$. 

0 200 400 600 800 1000
Figure 3. The dynamic regret of Algorithm 1 for problem (14) with constraint (15).

6. Conclusions

This paper introduces an online optimization algorithm for addressing distributed online aggregative problems featuring dynamic environments. With this algorithm, the dynamic regret converges to a boundary without relying on the condition that the dynamic environment is known. Future research includes the distributed online optimization problems in the context of time-varying directed networks.

Author Contributions: Conceptualization, C.Y., S.W. and B.H.; methodology, C.Y. and S.W.; writing—original draft preparation, C.Y. and S.W.; writing—review and editing, S.Z. and S.L.; funding acquisition, B.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Key R&D Program of China under Grant 2022ZD0119601.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Proof of Lemma 1. The proof follows from Lemma 2.1 in [11].
Appendix B

Proof of Lemma 2. Let \( e_{z_{i,t+1}} = \nabla \psi_i(x_{i,t+1}) - \nabla \psi_i(x_{i,t}) \). Then the algorithm (8f) can be rewritten as

\[
z_{i,t+1} = \sum_{j=1}^{N} W_{ij} \bar{z}_{j,t} + e_{z_{i,t+1}}.
\]

According to Lemma 2 in [12],

\[
\|z_{i,t+1} - \bar{z}_{t+1}\| \\
\leq N\gamma \beta^t \max_j \|z_{j,t}\| + \gamma \sum_{l=1}^{t-1} \beta^{t-l} \sum_{j=1}^{N} \|e_{z_{j,t+1}}\| + \frac{1}{N} \sum_{j=1}^{N} \|e_{z_{j,t+1}}\| + \|e_{z_{i,t+1}}\|. (A1)
\]

It follows from Assumption 2 that \( \|e_{z_{i,t+1}}\| \leq 2L \). Then,

\[
\|z_{i,t+1} - \bar{z}_{t+1}\| \leq \Delta_1.
\]

Moreover, it follows from (8f) that

\[
\frac{1}{N} \sum_{i=1}^{N} z_{i,t+1} = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{N} W_{ij} \bar{z}_{j,t} + \frac{1}{N} \sum_{i=1}^{N} \nabla \psi_i(x_{i,t+1}) - \frac{1}{N} \sum_{i=1}^{N} \nabla \psi_i(x_{i,t}).
\]

By using Assumption 1,

\[
z_{t+1} = \frac{1}{N} \sum_{i=1}^{N} \nabla \psi_i(x_{i,t+1}) = z_t - \frac{1}{N} \sum_{i=1}^{N} \nabla \psi_i(x_{i,t}),
\]

It follows from \( z_{i,0} = \nabla \psi_i(x_{i,0}) \) that \( \bar{z}_t = \frac{1}{N} \sum_{i=1}^{N} \nabla \psi_i(x_{i,t}) \). Assumption 2 implies that \( \|z_t\| \leq L \) holds. Therefore,

\[
\|\nabla_{l,t}\| \leq H + H(\|z_{i,t} - z_t\| + \|z_t\|) \\
\leq H + H\Delta_1 + HL = \Delta_2.
\]

\( \square \)

Appendix C

Proof of Lemma 3. Let \( e_{i,t} = \hat{x}_{i,t+1} - y_{i,t} \) and \( p_{i,t} = (A^T_l - I_n + \Delta A_{i,t})\hat{x}_{i,t+1} \). It follows from Lemma 2.2 in [11] that

\[
\|e_{i,t}\| \leq \Delta_2 \eta_t \\
\|p_{i,t}\| \leq (A_M Q + 8M Q \sum_{k=0}^{t} \lambda^{t-k}) \eta_t
\]

\[
\hat{x}_{t+1} = \frac{1}{N} \sum_{i=0}^{t} \sum_{j=1}^{N} (e_{j,r} + p_{j,t}),
\]

and

\[
\|x_{i,t+1} - \hat{x}_{t+1}\| \leq \sqrt{N} \left[ \|\nabla_{i,t}\| + M_{A} Q + \frac{8M Q}{1 - \lambda} \right] \sum_{r=0}^{t} \eta^{t-r}_r (W)\eta_r. (A3)
\]

By using the result \( \|\nabla_{i,t}\| \leq \Delta_2 \) in Lemma 2, (9) holds. \( \square \)
Appendix D

Proof of Lemma 4. Considering the first inequality in (10), and it follows from (A2) that

\[ \|x_{t+1} - x_t\| = \| \frac{1}{N} \sum_{j=1}^{N} (e_{j,t} + p_{j,t}) \| \]

\[ \leq (\Delta_2 + M_A Q + 8MQ \sum_{k=0}^{t} \lambda^{t-k}) \eta_t, \tag{A4} \]

which implies that

\[ \|x_{i,t+1} - x_{i,t}\| \]

\[ \leq \|x_{i,t+1} - \bar{x}_{t+1}\| + \|\bar{x}_{t+1} - \bar{x}_{t}\| + \|\bar{x}_{t} - x_{i,t}\| \]

\[ \leq (\Delta_2 + M_A Q + 8MQ \sum_{k=0}^{t} \lambda^{t-k}) \eta_t + 2\Delta_3 \sum_{\tau=0}^{t} c_2^{l-\tau} (W) \eta_{\tau}. \tag{A5} \]

Thus, the first inequality in (10) holds.

Consider the second inequality in (10). Note that there exists

\[ \|v(x_{i,t}) - v_{i,t}\| \]

\[ \leq \|v(x_{i,t}) - \frac{1}{N} \sum_{j=1}^{N} \psi_i(x_{j,t})\| + \| \frac{1}{N} \sum_{j=1}^{N} \psi_i(x_{j,t}) - v_{i,t}\| \]

\[ \leq \frac{L}{N} \sum_{j=1}^{N} \|x_{j,t} - x_{i,t}\| + \| \frac{1}{N} \sum_{j=1}^{N} \psi_i(x_{j,t}) - v_{i,t}\|, \tag{A6} \]

where the last line is since the function $\psi_i$ is Lipschitz (see Assumption 2). By using (9),

\[ \frac{L}{N} \sum_{j=1}^{N} \|x_{j,t} - x_{i,t}\| \]

\[ \leq \frac{L}{N} \sum_{j=1}^{N} (\|x_{i,t} - x_i\| + \|x_i - x_{j,t}\|) \]

\[ \leq 2L\Delta_3 \sum_{\tau=0}^{t} c_2^{l-\tau} (W) \eta_{\tau}. \tag{A7} \]

Let $e_{v_{j,t+1}} = \psi_i(x_{i,t+1}) - \psi_i(x_{j,t})$. By using Assumption 2 and (A5),

\[ \|e_{v_{j,t+1}}\| \leq L\|x_{i,t+1} - x_{j,t}\| \]

\[ \leq L(\Delta_2 + M_A Q + 8MQ \sum_{k=0}^{t} \lambda^{t-k}) \eta_t + 2\Delta_3 \sum_{\tau=0}^{t} c_2^{l-\tau} (W) \eta_{\tau}. \]

It follows from that

\[ v_{j,t+1} = \sum_{j=1}^{N} W_{ij}v_{j,t} + \psi_i(x_{i,t+1}) + e_{v_{j,t+1}}. \]

According to Lemma 2 in [12], it follows that
According to the initial state \( v_{i,0} = \psi_i(x_{i,0}) \), it follows that

\[
\frac{1}{N} \sum_{j=1}^{N} v_{i,j} = \frac{1}{N} \sum_{i=1}^{N} \psi_i(x_{i,j}).
\]

(A9)

It follows from (A8) and (A9) that

\[
\left\| \frac{1}{N} \sum_{i=1}^{N} \psi_i(x_{i,j}) - v_{i,j} \right\|
\leq N\gamma P^j \left( \frac{N\gamma \beta}{1 - \beta} + 2 \right) L(\Delta_2 + M_A Q + 8M \sum_{k=0}^{l-1} \lambda^{l-k}) \eta_{t-1} + (\frac{N\gamma \beta}{1 - \beta} + 2) 2L\Lambda_3 \sum_{\tau=0}^{t-1} \sigma_{2}^{l-1-\tau}(W)\eta_\tau.
\]

(A10)

Combining (A6), (A7), and (A10), the second inequality in (10) holds. Considering the third inequality in (10), and note that there exists

\[
\|z_{i,t} - \nabla v(x_{i,t})\|
\leq \|z_{i,t} - \frac{1}{N} \sum_{i=1}^{N} \nabla \psi_i(x_{i,t})\| + \| \frac{1}{N} \sum_{i=1}^{N} \nabla \psi_i(x_{i,t}) - \nabla v(x_{i,t})\|
\]

\[
\leq \|z_{i,t} - z_{i,t}\| + \frac{L}{N} \sum_{j=1}^{N} \|x_{i,t} - x_{j,t}\|.
\]

(A11)

Similar to (A8), there exists

\[
\|\varepsilon_{i,t+1}\| \leq \|x_{i,t+1} - x_{i,t}\|
\leq L \left( \Delta_2 + M_A Q + 8M \sum_{k=0}^{l-1} \lambda^{l-k} \right) \eta_t + 2L\Lambda_3 \sum_{\tau=0}^{t-1} \sigma_{2}^{l-1-\tau}(W)\eta_\tau.
\]

It follows from (A1) that

\[
\|z_{i,t} - z_{i,t}\| \leq N\gamma P^j \left( \frac{N\gamma \beta}{1 - \beta} + 2 \right) L(\Delta_2 + M_A Q + 8M \sum_{k=0}^{l-1} \lambda^{l-k}) \eta_t + (\frac{N\gamma \beta}{1 - \beta} + 2) 2L\Lambda_3 \sum_{\tau=0}^{t-1} \sigma_{2}^{l-1-\tau}(W)\eta_\tau.
\]

(A12)
Therefore, according to (A5), (A11), and (A12), the third inequality in (10) holds.

Appendix E

Proof of Lemma 5. The proof is follows from Lemma 2.3 in [11].

Appendix F

Proof of Lemma 6. It follows from Assumption 2 and Lemma 2 that
\[
\|\nabla_{i,t} - \nabla_{i,t}\|
\leq \|\nabla f_{i,t}(x_{i,t}, v(x_{i,t})) - \nabla f_{i,t}(x_{i,t}, v_{i,t})\|
+ \|\nabla f_{i,t}(x_{i,t}, v(x_{i,t}))\| \nabla v(x_{i,t}) - \nabla f_{i,t}(x_{i,t}, v_{i,t})z_{i,t}\|
\leq L\|v(x_{i,t}) - v_{i,t}\| + \|\nabla f_{i,t}(x_{i,t}, v(x_{i,t}))\| \nabla v(x_{i,t}) - \nabla f_{i,t}(x_{i,t}, v_{i,t})\|
\leq L\|v(x_{i,t}) - v_{i,t}\|\|\nabla v(x_{i,t})\|
\leq L^2 \{N\gamma P\beta^2 + \Delta_5 \eta_{i-1} + \Delta_6 \sum_{t=0}^{t} \sigma_1^{t-\tau}(W)\eta_t\},
\]

and
\[
\|\nabla f_{i,t}(x_{i,t}, v_{i,t})\| \nabla v(x_{i,t}) - \nabla f_{i,t}(x_{i,t}, v_{i,t})z_{i,t}\|
\leq \|\nabla f_{i,t}(x_{i,t}, v_{i,t})\| \|z_{i,t} - \nabla v(x_{i,t})\|
\leq H \{N\gamma P\beta^2 + \Delta_5 \eta_{i-1} + \Delta_6 \sum_{t=0}^{t} \sigma_1^{t-\tau}(W)\eta_t\}.
\]

Then the proof is over by combining (A13)–(A15) and (10).

Appendix G

Proof of Theorem 1. Note that the function $f_i$ is convex. Thus, there exists
\[
f_{i,t}(x_{i,t}) - f_{i,t}(x^*_i)
\leq \langle \nabla_{i,t}x_{i,t} - x^*_i \rangle
\]
\[
= \langle \nabla_{i,t}\hat{x}_{i,t+1} - x^*_i \rangle + \langle \nabla_{i,t}x_{i,t} - \hat{x}_{i,t+1} \rangle + \langle \nabla_{i,t}y_{i,t} - \hat{y}_{i,t+1} \rangle + \langle \nabla_{i,t}x_{i,t} - \hat{x}_{i,t+1} - x^*_i \rangle.
\]

According to algorithm (8a), Lemma 4.1 in [13], and the fact that $D_R(x, y) \geq \frac{1}{2\eta} \|x - y\|^2$ where $x, y \in \mathcal{X}$, it follows that
\[
\langle \nabla_{i,t}\hat{x}_{i,t+1} - x^*_i \rangle
\leq \frac{1}{\eta_t} D_R(x^*_i, y_{i,t}) - \frac{1}{\eta_t} D_R(x^*_i, \hat{x}_{i,t+1}) - \frac{1}{\eta_t} D_R(\hat{x}_{i,t+1}, y_{i,t})
\leq \frac{1}{\eta_t} D_R(x^*_i, y_{i,t}) - \frac{1}{\eta_t} D_R(x^*_i, \hat{x}_{i,t+1}) - \frac{1}{2\eta_t} \|\hat{x}_{i,t+1} - y_{i,t}\|^2.
\]
By using Lemmas 2 and 3, Assumption 1, and Algorithm (8d),

\begin{align*}
\langle \nabla_{i,t} x_{i,t} - y_{i,t} \rangle \\
= \langle \nabla_{i,t} x_{i,t} - x_t + x_t - y_{i,t} \rangle \\
= \langle \nabla_{i,t} x_{i,t} - x_t \rangle + \sum_{j=1}^{N} w_{ij} \langle \nabla_{i,t} x_t - x_{j,t} \rangle \\
\leq \| \nabla_{i,t} \| (\| x_{i,t} - x_t \| + \sum_{j=1}^{N} w_{ij} \| x_{j,t} - x_t \|) \\
\leq 2(H + HL) \Delta_3 \sum_{r=0}^{T} \sigma_2^{r-T} (W) \eta_r.
\end{align*}
(A18)

It follows from Lemma 2 that

\begin{align*}
\langle \nabla_{i,t} y_{i,t} - \hat{x}_{i,t+1} \rangle \\
\leq \| \nabla_{i,t} \| \| \hat{x}_{i,t+1} - y_{i,t} \| \\
\leq \frac{1}{2 \eta_t} \| \hat{x}_{i,t+1} - y_{i,t} \|^2 + \frac{(H + HL)^2}{2} \eta_t.
\end{align*}
(A19)

According to (A16)–(A19), we have that

\begin{align*}
f_{i,t}(x_{i,t}) - f_{i,t}(x^*_t) \\
\leq \langle \nabla_{i,t} - \nabla_{i,t} \hat{x}_{i,t} - x^*_t \rangle + \frac{1}{\eta_t} D_R(x^*_t, y_{i,t}) - \frac{1}{\eta_t} D_R(x^*_t, \hat{x}_{i,t+1}) \\
+ \frac{(H + HL)^2}{2} \eta_t + 2(H + HL) \Delta_3 \sum_{r=0}^{T} \sigma_2^{r-T} (W) \eta_r.
\end{align*}

Then it follows from Lemmas 5 and 6 that

\begin{align*}
&\sum_{t=1}^{T} \sum_{i=1}^{N} \left( f_{i,t}(x_{i,t}) - f_{i,t}(x^*_t) \right) \\
\leq & \sum_{t=1}^{T} \sum_{i=1}^{N} \| \hat{x}_{i,t} - x^*_t \| (L + L^2 + H) \{ N \gamma P \beta^t + \Delta_3 \eta_{t-1} + \Delta_3 \sum_{r=0}^{T} \sigma_2^{r-T} (W) \eta_r \} \\
+ & \frac{2NR^2}{\eta_{T+1}} + \sum_{t=1}^{T} \sum_{i=1}^{N} k_{i,t+1} \| x^*_t - x_{i,t} \|^2 + \sum_{t=1}^{T} \sum_{i=1}^{N} \frac{\eta_t}{\eta_{T+1}} \Delta_7 \| A^*_t x^*_t - x_{i,t+1} \| \\
+ & \frac{N(H + HL)^2}{2} \sum_{t=1}^{T} \eta_t + 2N(H + HL) \Delta_3 \sum_{r=0}^{T} \sum_{i=1}^{N} \sigma_2^{r-T} (W) \eta_r.
\end{align*}
(A20)

Note that \( \eta_T \) is fixed and satisfies \( \eta_t = \eta \). It follows from \( \| \hat{x}_{i,t} - x^*_t \| \leq 2Q \), (A20) and Assumption 2 that

\begin{align*}
&\sum_{t=1}^{T} \sum_{i=1}^{N} \| x_{i,t} - x^*_t \| \\
\leq & \Delta_7 \sum_{t=1}^{T} \sum_{i=1}^{N} \| x_{i,t} - x^*_t \| + \Delta_7 \frac{\eta T}{\eta} \sum_{i=1}^{T} \| \theta_i \| + \frac{2NR^2}{\eta} \\
+ & \sum_{i=1}^{N} \Delta_7 \| x^*_t - x_{i,T+1} \| + \Delta_7 \sum_{i=1}^{T} \| \theta_i \| + 2QN \gamma P (L + L^2 + H) \sum_{i=1}^{N} \beta^i.
\end{align*}
(A21)
Then it follows from (A21) that

\[
\sum_{t=1}^{T} \sum_{i=1}^{N} (t - \Delta_T) \|x_{i,t} - x^*_t\| \leq \Delta_T \eta T + \frac{2NR^2 + NKC_T}{\eta} + \Delta_9.
\]  

(A22)

Therefore, the proof is over by combining (A22) with

\[
\text{Reg}(T) \leq \sum_{t=1}^{T} \sum_{i=1}^{N} L \|x_{i,t} - x^*_t\|.
\]

References


2. Li, X.X.; Xie, L.H.; Li, N. A survey on distributed online optimization and online games. *Annu. Rev. Control* 2023, 56, 24. [CrossRef]


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.