Lower Bounds on Multivariate Higher Order Derivatives of Differential Entropy †

Laigang Guo 1, Chun-Ming Yuan 2,3 and Xiao-Shan Gao 2,3,*

1 Laboratory of Mathematics and Complex Systems, Ministry of Education, School of Mathematical Sciences, Beijing Normal University, Beijing 100875, China
2 KLMM, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China
3 University of Chinese Academy of Sciences, Beijing 100049, China
* Correspondence: xgao@mmrc.iss.ac.cn
† This paper is an extended version of our paper published in 2021 IEEE International Symposium on Information Theory.

Abstract: This paper studies the properties of the derivatives of differential entropy $H(X_t)$ in Costa’s entropy power inequality for real-valued random variables. Cheng and Geng conjectured that for $m \geq 1$, $(-1)^m + (d^m/dt^m) H(X_t) \geq 0$, while McKean conjectured a stronger statement, whereby $(-1)^m + (d^m/dt^m) H(X_t) \geq (-1)^m + (d^m/dt^m) H(X_{Gt})$. Here, we study the higher dimensional analogues of these conjectures. In particular, we study the veracity of the following two statements: $C_1(m, n) : (-1)^m + (d^m/dt^m) H(X_t) \geq 0$, where $n$ denotes that $X_t$ is a random vector taking values in $\mathbb{R}^n$, and similarly, $C_2(m, n) : (-1)^m + (d^m/dt^m) H(X_t) \geq (-1)^m + (d^m/dt^m) H(X_{Gt}) \geq 0$. In this paper, we prove some new multivariate cases: $C_3(3, i), i = 2, 3, 4$. Motivated by our results, we further propose a weaker version of McKean’s conjecture $C_4(m, n) : (-1)^m + (d^m/dt^m) H(X_t) \geq (-1)^m + (d^m/dt^m) H(X_{Gt})$, which is implied by $C_2(m, n)$ and implies $C_1(m, n)$. We prove some multivariate cases of this conjecture under the log-concave condition: $C_5(3, i), i = 2, 3, 4$ and $C_6(4, 2)$. A systematic procedure to prove $C_1(m, n)$ is proposed based on symbolic computation and semidefinite programming, and all the new results mentioned above are explicitly and strictly proved using this procedure.

Keywords: differential entropy; completely monotone; McKean’s conjecture; log-concavity; Gaussian optimality

1. Introduction

Shannon’s entropy power inequality (EPI) is one of the most important information inequalities [1], which has many proofs, generalizations, and applications [2–11]. In particular, Costa presented a generalized version of the EPI in his seminal paper [12].

Let $X$ be an $n$-dimensional random vector with finite variance and a probability density function $p(x)$. For $t > 0$, define $X_t \triangleq X + Z_t$, where $Z_t \sim N_0(0, tI)$ is an independent standard Gaussian random vector with the covariance matrix $t \times I$. The probability density of $X_t$ is

$$p_t(x_t) = \frac{1}{(2\pi t)^{n/2}} \int_{\mathbb{R}^n} p(x) \exp \left( -\frac{\|x_t - x\|^2}{2t} \right) dx. \quad (1)$$

Thus, the heat equation holds for $p_t(x_t)$, i.e.,

$$\frac{dp_t}{dt} = \frac{1}{2} \nabla^2 p_t. \quad (2)$$

**The differential entropy** of $X_t$ is defined as

$$H(X_t) = -\int_{\mathbb{R}^n} p_t(x_t) \log p_t(x_t) dx_t. \quad (3)$$
Costa [12] proved that the entropy power of $X_t$, given by $N(X_t) = \frac{1}{2\pi e} e^{(2/n)H(X_t)}$, is a concave function in $t$. More precisely, Costa proved $(d/dt)N(X_t) \geq 0$ and $(d^2/dt^2)N(X_t) \leq 0$.

Due to its importance, several new proofs and generalizations for Costa’s EPI have been given. Dembo [13] gave a simple proof for Costa’s EPI via the Fisher information inequality. Villani [14] proved Costa’s EPI with Cauchy–Schwarz inequality as well as the heat equation. Toscani [15] proved that $(d^3/dt^3)N(X_t) \geq 0$ if $p_t$ is log-concave. Cheng and Geng proposed a conjecture [16]:

**Conjecture 1.** The first derivative of $H(X_t)$ (i.e., the Fisher information) is completely monotone in $t$, that is,

$$C_1(m, n) := (-1)^{m+1}(d^m / dt^m)H(X_t) \geq 0.$$  \hspace{1cm} (4)

Costa’s EPI implies $C_1(1, n)$ and $C_1(2, n)$ [12], and Cheng–Geng proved $C_1(3, 1)$ and $C_1(4, 1)$ [16].

Let $X_G \sim N_n(\mu, \sigma^2 I)$ be an $n$-dimensional Gaussian random vector and $X_{Gt} \equiv X_G + Z_t$ be the Gaussian $X_t$. McKean [17] proved that $X_{Gt}$ achieves the minimum of $(d/dt)H(X_t)$ and $- (d^2/dt^2)H(X_t)$ is subject to $\text{Var}(X_t) = \sigma^2 + t$, and conjectured the general case:

**Conjecture 2.** The following inequality holds subject to $\text{Var}(X_t) = \sigma^2 + t$,

$$C_2(m, n) := (-1)^{m+1}(d^m / dt^m)H(X_t) \geq (-1)^{m+1}(d^m / dt^m)H(X_{Gt}) \geq 0.$$  \hspace{1cm} (5)

McKean proved $C_2(1, 1)$ and $C_2(2, 1)$ [17]. Zhang–Anantharam–Geng [18] proved $C_2(3, 1)$, $C_2(4, 1)$ and $C_2(5, 1)$ if the probability density function of $X_t$ is log-concave. Note that $C_2(1, n)$ and $C_2(2, n)$ are immediate consequences of Entropy Power Inequality and Costa’s concavity of entropy power result [12], respectively. In this paper, we notice that in the multivariate case, Conjecture 2 might not be true for $m > 2$ even under the log-concave condition, which motivates us to propose the following weaker conjecture:

**Conjecture 3.** The following inequality holds subject to $\text{Var}(X_t) = \sigma^2 + t$,

$$C_3(m, n) := (-1)^{m+1}(d^m / dt^m)H(X_t) \geq (-1)^{m+1}(d^m / dt^m)H(X_{Gt}) \geq 0.$$  \hspace{1cm} (6)

We see that Conjecture 3 coincides with Conjecture 2 for $n = 1$ (univariate case). Additionally, Conjecture 2 implies Conjecture 3 and Conjecture 3 implies Conjecture 1. The three conjectures give different lower bounds for the derivatives of $(-1)^{m+1}H(X_t)$.

**Remark 1.** The authors in [14,16] proved some cases of Conjecture 1 by writing the left-hand formula in Conjecture 1 as sums of squares and, hence, concluded their sign. We provide a systematic way to explore this idea using symbolic computation and semidefinite programming and prove several new results in the multivariate cases.

Our procedure for proving $C_3(m, n)$ consists of three main ingredients. First, a systematic method is proposed to compute the constraints $R_i, i = 1, \ldots, N_1$ that are satisfied by $p_t(x_t)$ and its derivatives. The condition that $p_t$ is log-concave can also be reduced to a set of constraints, i.e., $R_{ij}, j = 1, \ldots, N_2$. Second, based on symbolic computation, proof for $C_3(m, n)$ is reduced to the following problem:

$$\exists p_t \in \mathbb{R} \text{ and } Q_j \text{ s.t. } (E - \sum_{i=1}^{N_1} p_t R_i - \sum_{j=1}^{N_2} Q_j R_j = S)$$  \hspace{1cm} (7)

where $E, Q_j,$ and $S$ are polynomials in $p_t$ and its derivatives such that $E$ represents the conjecture, $Q_j \geq 0,$ and $S$ is a sum of squares (SOS). Third, problem (7) can be solved with semidefinite programming (SDP) [19,20]. Note that from Equation (7), we can give an explicit and strict proof for $C_3(m, n)$.
Using the procedure proposed in this paper, we prove several new results about the three conjectures: \(C_1(3, 2), C_1(3, 3), C_1(3, 4), \) and \(C_5(3, 2), C_5(3, 3), C_5(3, 4), C_5(4, 2)\) under the log-concave condition.

In Table 1, we give the data for computing the SOS representation (7) using the Matlab software in Appendix A of [21], where \(N_1\) and \(N_2\) are the numbers of constraints in (7).

### Table 1. Data in computing the SOS with symbolic computation and SDP.

|          | \(C_2(3, 1)\) | \(C_1(3, 2)\) | \(C_1(3, 3)\) | \(C_1(3, 4)\) | \(C_3(3, 2)\) | \(C_3(3, 3)\) | \(C_3(3, 4)\) | \(C_5(4, 2)\) |
|----------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| Vars     | 3             | 14            | 38            | 80            | 14            | 38            | 38            | 33            |
| \(N_1\)  | 6             | 63            | 512           | 1966          | 63            | 512           | 512           | 417           |
| \(N_2\)  | 0             | 0             | 0             | 0             | 6             | 6             | 3             |               |

The procedure is inspired by the work of [12,14,16,18], and uses basic ideas introduced therein. The specific contributions in this paper are:

1. Based on symbolic computation and semidefinite programming, \(C_s(m, n)\) can be automatically verified with the aid of the software systems Maple and Matlab, and analytical proofs for \(C_s(m, n)\) can also be efficiently produced.

2. The new concept of differentially homogenous polynomials is introduced and used to reduce the computational complexity. Compared with the pure SDP-based approach (such as [18]), the computational efficiency of our procedure is, in general, much higher. See Procedure 2 for details.

3. The results in [16,18] are generalized from the univariate cases to the multivariate cases (new results). This is the first attempt for the multivariate high order cases of the conjectures.

4. In comparison to the literature (such as [12,15,16,18]), the constraints (integral or log-concave) considered in this paper are more general.

The rest of this paper is organized as follows. In Section 2, we give the proof procedure. In Section 3, we prove \(C_1(3, 2), C_1(3, 3)\) and \(C_1(3, 4)\). In Section 4 we prove \(C_3(3, 2), C_3(3, 3)\), and \(C_3(3, 4)\) under the log-concave condition. In Section 5, we prove \(C_5(4, 2)\) under the log-concave condition. In Section 6, the conclusions are presented.

### 2. Proof Procedure

In this section, we provide a general procedure to prove \(C_s(m, n)\) for specific values of \(s, m,\) and \(n\).

#### 2.1. Some Notations

Let \([n]_0 = \{0, 1, \ldots, n\}\), \([n] = \{1, \ldots, n\}\), and \(x_i = [x_{1,i}, \ldots, x_{n,i}]\). To simplify the notations, we use \(p_i\) to denote \(p_i(x_i)\) in the rest of the paper. Denote

\[
\mathcal{P}_n = \left\{ \frac{\partial^n p_i}{\partial x_{1,i}^{\delta_1} \cdots \partial x_{n,i}^{\delta_n}} : h = \sum_{i=1}^{n} h_i, h_i \in \mathbb{N} \right\}
\]

to be the set of all derivatives of \(p_i\) with respect to the differential operators \(\frac{\partial}{\partial x_{1,i}}, i = 1, \ldots, n\) and \(\mathbb{R}[\mathcal{P}_n]\) to be the set of polynomials in \(\mathcal{P}_n\) with coefficients in \(\mathbb{R}\). For \(v \in \mathcal{P}_n\), let \(\text{ord}(v)\) be the order of \(v\). For a monomial \(\prod_{i=1}^{r} v_{i}^{d_i}\) with \(v_i \in \mathcal{P}_n\), its degree, order, and total order are defined as \(\sum_{i=1}^{r} d_i\), \(\max_{i=1}^{r} \text{ord}(v_i)\), and \(\sum_{i=1}^{r} d_i \cdot \text{ord}(v_i)\), respectively.

A polynomial in \(\mathbb{R}[\mathcal{P}_n]\) is called a \(k\)-th order differentially homogeneous polynomial or simply a \(k\)-th order differential form, if all its monomials have a degree of \(k\) and a total order of \(k\). Let \(\mathcal{M}_{k,n}\) be the set of all monomials which have a degree of \(k\) and a total order of \(k\). Then, the set of \(k\)-th order differential forms is an \(\mathbb{R}\)-linear vector space generated by \(\mathcal{M}_{k,n}\), which is denoted as \(\text{Span}_{\mathbb{R}}(\mathcal{M}_{k,n})\).
We will use Gaussian elimination in Span$_R(M_{k,n})$ by treating the monomials as variables. We always use the lexicographic order for the monomials to be defined below unless mentioned otherwise. Consider two distinct derivatives $v_1 = \frac{\partial^2 p}{\partial x_{i_1} \partial x_{i_2}}$ and $v_2 = \frac{\partial^2 p}{\partial x_{j_1} \partial x_{j_2}}$. We say $v_1 > v_2$ if $h > s$, or $h = s, h_1 > s_1$ and $j = s$ for $j = l + 1, \ldots, n$.

Consider the two distinct monomials $m_1 = \prod_{i=1}^s v_i^{l_i}$ and $m_2 = \prod_{i=1}^s v_i^{r_i}$, where $v_i \in P_n$ and $v_i < v_j$ for $i < j$. We define $m_1 > m_2$ if $d_i > e_i$, and $d_i = e_i$ for $i = l + 1, \ldots, r$.

From (1), $p_t : R^{n+1} \rightarrow R$ is a function in $x_i$ and $t$. Therefore, each polynomial $f \in R[|P_n|]$ is also a function in $x_i$ and $t$, $f(t) = \int_{R^n} f dx_i$ is a function in $t$, and the expectation of $f$ with respect to $x_i$ $E[f] \triangleq \int_{R^n} p_t f dx_i$ is also a function in $t$. By $f \geq 0, f \geq 0$, and $E[f] \geq 0$, we mean $f(x_i, t) \geq 0$, $\tilde{f}(t) \geq 0$, and $E[f](t) \geq 0$ for all $x_i \in R^n$ and $t > 0$.

2.2. Three Parts of the Proof

In this section, we give the procedure to prove $C_s(m, n)$, which consists of three parts.

2.2.1. Part I

In step 1, we reduce the proof of $C_s(m, n)$ into the proof of an integral inequality, as shown by the following lemma, whose proof will be given in Section 2.3:

**Lemma 1.** Proof that $C_s(m, n), s = 1, 2, 3$ can be reduced to show

$$\int_{R^n} \frac{E_{s,m,n}}{p_t^{2m-1}} dx_i \geq 0 \quad (8)$$

where

$$E_{s,m,n} = \sum_{a_1=1}^n \cdots \sum_{a_m=1}^n E_{s,m,n,a_m},$$

$$a_m = (a_1, \ldots, a_m),$$

$E_{s,m,n,a_m}$ is a $2m$th-order differential form in $R[|P_{m,n}|]$, and

$$P_{m,n} = \{ h \in [2m-1]: a_i \in [n], i \in [m] \}. \quad (9)$$

2.2.2. Part II

In step 2, we compute the constraints which are relations satisfied by the probability density $p_t$ of $X_i$. In this paper, we consider two types of constraints: integral constraints and log-concave constraints, which will be given in Lemmas 2 and 3, respectively. Since $E_{s,m,n}$ in (8) is a $2m$th-order differential form, we need only the constraints which are $2m$th-order differential forms.

**Definition 1.** An $m$th-order integral constraint is the $2m$th-order differential form $R$ in $R[|P_n|]$ such that

$$\int_{R^n} \frac{R}{p_t^{2m-1}} dx_i = 0.$$

**Lemma 2 ([22]).** There is a systematic method to compute the $m$th-order integral constraints $C_{m,n} = \{ R_i, i = 1, \ldots, N_1 \}$.

A function $f : R^n \rightarrow R$ is called log-concave if $\log f$ is a concave function. In this paper, by the log-concave condition, we mean that the density function $p_t$ is log-concave.

**Definition 2.** An $m$th-order log-concave constraint is a $2m$th-order differential form $R$ in $R[|P_n|]$ such that $R \geq 0$ under the log-concave condition.

The following lemma computes the log-concave constraints:
Lemma 3 ([22]). Let $H(p_t) \in \mathbb{R}[^n x n]$ be the Hessian matrix of $p_t$, $\nabla p_t = (\frac{\partial p_t}{\partial x_{ij}}, \ldots, \frac{\partial p_t}{\partial x_{nm}})$,

$$L(p_t) \triangleq p_t H(p_t) - \nabla^T p_t \nabla p_t,$$

and $\triangle_{k,i}, l = 1, \ldots, L_k$ be the $k$th-order principle minors of $L(p_t)$. Then, the $m$th-order log-concave constraints are

$$C_{m,n} = \{ \prod_{i=1}^{l} (-1)^{k_i} \triangle_{k_i,i} | \sum_{i=1}^{l} k_i \leq m \}$$

where $T_{k_1,\ldots,k_l} \in \text{Span}_{\mathbb{R}}(\mathcal{M}_{2m-2\sum_{i=1}^{l} k_i,n})$ and $T_{k_1,\ldots,k_l} \geq 0$.

Note that $T_{k_1,\ldots,k_l}$ in (11) are not known. For convenience, denote

$$C_{m,n} = \{ P_j, j = 1, \ldots, N_2 \},$$

where $P_j$ represents $\prod_{i=1}^{l} (-1)^{k_i} \triangle_{k_i,i}$ in (11). From Lemma 3, it is easy to see that $\prod_{i=1}^{l} (-1)^{k_i} \triangle_{k_i,i}$ is a $(2\sum_{i=1}^{l} k_i)$th-order log-concave constraint.

2.2.3. Part III

In step 3, we give a procedure to write $E_{s,m,n}$ as an SOS under the constraints, the details of which will be given in Section 2.4.

Procedure 1. For $E_{s,m,n}$ in Lemma 1, $C_{m,n} = \{ R_i, i = 1, \ldots, N_1 \} \in \text{Lemma 2}$, and $C_{m,n} = \{ P_j, j = 1, \ldots, N_2 \}$ in Lemma 3, the procedure computes $e_t \in \mathbb{R}$ and $Q_j \in \text{Span}_{\mathbb{R}}(\mathcal{M}_{2m-\deg p_j,n})$ such that

$$E_{s,m,n} - \sum_{i=1}^{N_1} e_t R_i - \sum_{j=1}^{N_2} P_j Q_j = S,$$

and $Q_j \geq 0, j = 1, \ldots, N_2$,

where $S$ is an SOS. If the log-concave condition is not needed, we may set $Q_j = 0$ for all $j$.

To summarize the proof procedure, we have the following:

Theorem 1. If Procedure 1 satisfies (13) and (14) for certain $s, m,$ and $n$, then $C_{s}(m,n)$ is explicitly and strictly proved.

Proof. With Lemma 1, we have the following proof for $C_{s}(m,n)$:

$$\int_{\mathbb{R}} \frac{E_{s,m,n}}{p_t^{2m-1}} \, dx_t \overset{(13)}{=} \int_{\mathbb{R}} \frac{\sum_{i=1}^{N_1} e_t R_i + \sum_{j=1}^{N_2} P_j Q_j + S}{p_t^{2m-1}} \, dx_t$$

$$\overset{S1}{=} \int_{\mathbb{R}} \frac{\sum_{j=1}^{N_2} P_j Q_j + S}{p_t^{2m-1}} \, dx_t$$

$$\overset{S2}{\geq} \int_{\mathbb{R}} \frac{S}{p_t^{2m-1}} \, dx_t$$

$$\overset{S3}{\geq} 0.$$

Equality S1 is true, because $R_i$ is an integral constraint by Lemma 2. By Lemma 3 and (14), $P_j Q_j \geq 0$ is true under the log-concave condition, so inequality S2 is true under the log-concave condition. Finally, inequality S3 is true, because $S \geq 0$ is an SOS. \qed
2.3. Proof of Lemma 1

Costa [12] proved the following basic properties for $p_t$ and $H(X_t)$,

$$\frac{dH(X_t)}{dt} = -\frac{1}{2} \mathbb{E} \left[ \nabla^2 \log p_t \right]$$

$$= \frac{1}{2} \int_{\mathbb{R}^n} \frac{\| \nabla p_t \|^2}{p_t} dx_t$$

(16)

where

$$\nabla p_t = \left( \frac{\partial p_t}{\partial x_{1,j}}, \ldots, \frac{\partial p_t}{\partial x_{n,j}} \right), \nabla^2 p_t = \sum_{i=1}^n \frac{\partial^2 p_t}{\partial^2 x_{i,j}},$$

and $J(X_t) \triangleq \mathbb{E} \left( \| \nabla p_t \|^2 \right)$ is the Fisher information [6]. Equation (16) implies $C_1(1, n)$: $\frac{d}{dt} H(X_t) \geq 0$.

For $s = 1$, Lemma 1 was proved by

**Lemma 4 ([22]).** For $m \in \mathbb{N}_{m>1}$, we have

$$(-1)^{m+1} \left( \frac{d^m}{dt^m} \right) H(X_t) = \int_{\mathbb{R}^n} \frac{E_{1,m,n}}{p_t^{2m-1}} dx_t,$$

(17)

where

$$E_{1,m,n} = \frac{(-1)^{m+1} p_t^{2m-1}}{2} \frac{d^{m-1} \| \nabla p_t \|^2}{dt^{m-1}} \frac{p_t}{p_t}$$

$$= \sum_{a_1=1}^n \cdots \sum_{a_m=1}^n E_{1,m,n,a_m}$$

is a 2mth-order differential form in $\mathbb{R}[\mathcal{P}_{m,n}]$.

To prove Lemma 1 for $s = 2, 3$, we need to compute $(d^m/dt^m)H(X_{Gt})$. Let $X_G \sim N_n(\mu, \sigma^2 I)$ be an $n$-dimensional Gaussian random vector and $X_{Gt} \triangleq X_G + Z_t$, where $Z_t \sim N_n(0, tI)$ is introduced in Section 1. Then, $X_{Gt} \sim N_n(\mu, (\sigma^2 + t) I)$ and the probability density of $X_{Gt}$ is

$$\hat{p}_t = \frac{1}{(2\pi(\sigma^2 + t))^{n/2}} \exp \left( -\frac{1}{2(\sigma^2 + t)} \| x_t - \mu \|^2 \right).$$

**Lemma 5 ([22]).** Let $T = \nabla^2 \log p_t$ and $T_G = \nabla^2 \log \hat{p}_t$. Then, under the log-concave condition, we have

$$E[(-T)^m] \overset{(a)}{\geq} |E(-T)|^m \overset{(b)}{\geq} |E(-T_G)|^m \overset{(c)}{=} (-1)^{m+1} 2^{m-1} \left( \frac{d^m}{dt^m} \right) H(X_{Gt}).$$

(18)

**Lemma 6 ([22]).** For $T = \nabla^2 \log p_t$ and $m \in \mathbb{N}_{m>1}$, we have

$$E[(-T)^m] = \int_{\mathbb{R}^n} E_{0,m,n} dX_t,$$

(19)

where

$$E_{0,m,n} = \sum_{a_1=1}^n \cdots \sum_{a_m=1}^n E_{0,m,n,a_m},$$

$$a_m = (a_1, \ldots, a_m),$$

and $E_{0,m,n,a_m}$ is a 2mth-order differential form in $\mathbb{R}[\mathcal{P}_{m,n}]$. 


We can now prove Lemma 1 for $s = 2, 3$. Let

$$E_{2,m,n} = E_{1,m,n} - \frac{(m-1)!}{2^m} E_{0,m,n},$$

$$E_{3,m,n} = E_{1,m,n} - \frac{(m-1)!}{2^m} E_{0,m,n},$$

(20)

where $E_{1,m,n}$ and $E_{0,m,n}$ are from Lemmas 4 and 6, respectively. By Lemma 5, $C_s(m,n)$ is true if $\int_{\mathbb{R}^n} \frac{E_{2,m,n}}{p_1} dx_i \geq 0$ for $i = 2, 3$. Together with Lemma 4, Lemma 1 is proved.

### 2.4. Main Result (Procedure 1)

In this section, we present the detailed Procedure 1, called Procedure 2, which is based on symbolic computation and the SOS theory.

**Procedure 2. Input:** $E_{s,m,n}$ and $R_i, i = 1, \ldots, N_1$ are $2m$th-order differential forms in $\mathbb{R}[\mathcal{P}_n]$; $P_j, j = 1, \ldots, N_2$ are $2k_i$th-order differential forms in $\mathbb{R}[\mathcal{P}_n]$.

**Output:** $e_i \in \mathbb{R}$ and $Q_j \in \text{Span}_\mathbb{R}(\mathcal{M}_{2(m-k_i)}, n)$ such that (13) and (14) are true, or fail meaning such that $e_i$ and $Q_j$ are not found.

**S1.** Treat the monomials in $\mathcal{M}_{m,n}$ as new variables $m_l, l = 1, \ldots, N_{m,n}$, which are all the monomials in $\mathbb{R}[\mathcal{P}_n]$ with the degree $m$ and the total order $m$. We call $m_l m_i$ a quadratic monomial.

**S2.** Write monomials in $C_{m,n} = \{R_i, i = 1, \ldots, N_1\}$ as quadratic monomials if possible. By performing Gaussian elimination on $C_{m,n}$ by treating the monomials as variables and according to a monomial order such that a quadratic monomial is less than a non-quadratic monomial, we obtain

$$\tilde{C}_{m,n} = C_{m,n,1} \cup C_{m,n,2},$$

where $C_{m,n,1}$ is the set of quadratic forms in $m_l, C_{m,n,2}$ is the set of non-quadratic forms, and $\text{Span}_\mathbb{R}(C_{m,n}) = \text{Span}_\mathbb{R}(\tilde{C}_{m,n})$.

**S3.** There may exist relationships among the variables $m_l$, which are called intrinsic constraints. For instance, for $m_1 = \frac{p_i^2}{\sum_{x_{i,j}}} + m_2 = p_i \frac{\partial m_1}{\partial x_{i,j}} + \frac{\partial p_i}{\partial x_{i,j}}$, and $m_3 = \frac{\partial p_i}{\partial x_{i,j}}$ in $\mathcal{M}_4, n$, an intrinsic constraint is $m_1 m_3^2 = m_2^2 = 0$. By adding the intrinsic constraints which are quadratic forms in $m_l$ to $C_{m,n,1,}$, we obtain

$$\tilde{C}_{m,n,1} = \{\tilde{R}_i, i = 1, \ldots, N_3\}.$$

**S4.** Let $\mathcal{M}_{2(m-k_i)}, n = \{m_{l,k}, k = 1, \ldots, V_j\}$ and $Q_j = \sum_{k=1}^{V_j} q_{j,k} m_{j,k}$, where $q_{j,k}$ are variables to be found later. Let $\tilde{R}_j$ be obtained from $P_j Q_j$ by writing monomials in $P_j Q_j$ as quadratic monomials in $m_{l,k}$, and eliminating the non-quadratic monomials with $C_{m,n,2}$ such that $\mathcal{R}_j - P_j Q_j \in \text{Span}_\mathbb{R}(C_{m,n})$ and $\mathcal{R}_j = \sum_{k=1}^{V_j} q_{j,k} h_{j,k}$, where $h_{j,k} \in \mathbb{R}[m_l, \mathcal{P}_n]$. If an $h_{j,k}$ is not a quadratic form in $m_l$, then delete $\tilde{R}_j$; hence, the $\tilde{R}_j$’s in quadratic form are selected. Then, denote these constraints as $R_j, j = 1, \ldots, N_2$, which form the reduced set $\tilde{C}_{m,n}$.

**S5.** Let $\tilde{E}_{s,m,n}$ be obtained from $E_{s,m,n}$ by eliminating the non-quadratic monomials using $C_{m,n,2}$ such that $E_{s,m,n} - \tilde{E}_{s,m,n} \in \text{Span}_\mathbb{R}(C_{m,n,2}) \subset \text{Span}_\mathbb{R}(C_{m,n})$.

**S6.** Since $\tilde{E}_{s,m,n}, \tilde{R}_i, i = 1, \ldots, N_3$ and $R_j, j = 1, \ldots, N_2$ are quadratic forms in $m_l$, we can use the Matlab codes given in Appendix A [21] to compute $p_i, q_{j,k} \in \mathbb{R}$ such that

$$\tilde{E}_{s,m,n} - \sum_{i=1}^{N_3} p_i \tilde{R}_i - \sum_{j=1}^{N_2} R_j = S,$$

(21)

$$R_j = \sum_{i=1}^{V_j} q_{j,i} h_{j,i}, j = 1, \ldots, N_2$$

$$Q_j = \sum_{i=1}^{V_j} q_{j,i} m_{j,i} \geq 0, j = 1, \ldots, N_2$$

(22)
where

\[ S = \sum_{i=1}^{N_{m,n}} \sum_{j=1}^{N_{m,n}} c_i (\sum_{j=1}^{e_j} m_j)^2 \]

is an SOS, \( c_i, e_j \in \mathbb{R} \) and \( c_i \geq 0 \). If (21) and (22) cannot be found, return FAIL.

**S7.** Since \( \hat{R}_i, \hat{E}_{s,m,n} - \hat{E}_{s,m,n}, R_j - P_i Q_j \) are all in \( \text{Span}_{\mathbb{R}}(C_{m,n}) \), Equations (13) and (14) can be obtained from (21) and (22), respectively.

**Remark 2.** Procedure 2 can be implemented automatically by Maple and Matlab on a computer. In Procedure 2, steps S2, S4 and S5 are based on the symbolic computation theory for reduction, which makes our method more efficient than the pure SDP-based method [18] or a direct theoretical proof [16]. The use of symbolic computation also ensures that our calculation is strict and free of numerical errors.

**Remark 3.** Let \( R \) be an intrinsic constraint. Then, \( R \) becomes zero when replacing \( m_i \) by its corresponding monomial in \( M_{m,n} \). Therefore, \( \text{Span}_{\mathbb{R}}(\hat{C}_{m,n,1}) = \text{Span}_{\mathbb{R}}(C_{m,n,1}) \subset \text{Span}_{\mathbb{R}}(\hat{C}_{m,n}) \) in \( \mathbb{R}[P_n] \); that is, we do not need to include the intrinsic constraints in (21). However, these intrinsic constraints are needed when using the Matlab software in Appendix A of [21].

### 2.5. An Illustrative Example

As an illustrative example, we prove \( C_2(3,1) \) under the log-concave condition using the proof procedure given in Section 2.2. Since \( n = 1 \), denote

\[
x_t = x_{1,t}, f := f_0 := p, f_n := \frac{\partial^n \varphi}{\partial x_{1,t}^n}, n \in \mathbb{N}_{>0}.
\]

In step 1, by Lemma 1 and (8), we have

\[
\begin{align*}
\frac{d^3 H(X_t)}{dt^3} - \frac{21}{2} E \left[ \frac{(f_1^2 - f f_2)^3}{f^6} \right] \\
= \int \left( \frac{2}{3} f_1 \left[ f_1^2 \right] - \frac{(f_1^2 - f f_2)^3}{f^6} \right) dx_t \\
= \int E_{2,3,1} dx_t
\end{align*}
\]

(23)

where

\[
E_{2,3,1} = \frac{1}{4} f^4 f_2^2 - \frac{1}{2} f_1^3 f_1 f_3 f_2 + \frac{1}{4} f_1^4 f_1 f_5 - \frac{11}{4} f_2 f_1^2 f_2^2
\]

\[
- \frac{1}{8} f_1^2 f_4 + f_3^3 f_2^3 + 3 f_1^4 f_1 f_2 - f_1^6
\]

is a sixth-order differential form.

In step 2, we compute the constraints with Lemmas 2 and 3. With Lemma 2, we find six third-order integral constraints: \( C_{3,1} = \{ R_i, i = 1, \ldots, 6 \} \):

\[
\begin{align*}
R_1 &= 5 f f_1^4 f_2 - 4 f_1^6, \\
R_2 &= 2 f_1^3 f_1 f_2 f_3 + f_1^2 f_2^2 - 2 f_2 f_1 f_2^2 f_2, \\
R_3 &= f_1^4 f_1 f_5 + f_1^3 f_2 f_4 - f_1^3 f_2^2 f_4, \\
R_4 &= f_1^2 f_4 + 2 f f_1 f_2 f_3 - 2 f_2 f_1^2 f_2, \\
R_5 &= f_1^4 f_1 f_3 + 3 f_1^2 f_1 f_2^2 - 3 f_1 f_2 f_2, \\
R_6 &= f_1^4 f_1 f_4 + f_1^4 f_2^2 - f_1^3 f_1 f_2 f_3.
\end{align*}
\]

With Lemma 3, we obtain one third-order log-concave constraint: \( C_{3,1} = \{ P_1 Q_1 \} \), where

\[
P_1 = f f_2 - f_1^2, Q_1 \in \text{Span}_{\mathbb{R}}(M_{4,1}), \text{ and } Q_1 \geq 0.
\]
In step 3, we use Procedure 2 to compute the SOS representation (13) and (14) with the input $E_{2,3,1}, C_{3,1} = \{ R_i, i = 1, \ldots, 6\}, P_1 = f_1^2 - f_2$.

S1. The new variables are $M_{3,1} = \{ m_1 = f_2^2 f_3, m_2 = f f_1 f_2, m_3 = f_1^3 \}$, which are listed from high to low in the lexicographical monomial order.

S2. By writing monomials in $C_{3,1}$ as quadratic monomials in $m_i$ if possible and performing Gaussian elimination on $C_{3,1}$, we have

$$\begin{align*}
C_{3,1,1} &= \{ \tilde{R}_1 = 5m_2 m_3 - 4m_2^3, \\
             &\quad \tilde{R}_2 = m_1 m_3 + 3m_2^2 - \frac{13}{2} m_3^2 \}, \\
C_{3,1,2} &= \{ \tilde{R}_1 = f_1^3 f_2^2 + 2m_1 m_2 - 2m_2^3, \\
             &\quad \tilde{R}_2 = f_1^4 f_1 f_2 - m_1^3 + 3m_2 m_3 + 6m_3^2 - \frac{24}{5} m_2^3, \\
             &\quad \tilde{R}_3 = f_1^4 f_2 f_4 + m_2^3 - m_1 m_2, \\
             &\quad \tilde{R}_4 = f_1^2 f_2^3 f_4 + 2m_1 m_2 + 6m_2^2 - \frac{24}{5} m_3^2 \}.
\end{align*}$$

S3. There exist no intrinsic constraints and thus, $\tilde{C}_{3,1,1} = \{ \tilde{R}_1, \tilde{R}_2 \}$ and $N_3 = 2$.

S4. $M_{4,1} = \{ f_1^3 f_4, f_1^2 f_3 f_2, f_2^2 f_3, f f_1 f_2 f_3, f_1^4 \}$. Then, $Q_1 = q_{1,1} f_2 f_3^2 + q_{1,2} f_3 f_2^2 + q_{1,3} f_1^4$.

Monomials $f_3^3 f_4, f_2^3 f_1 f_3$ do not appear in $Q_1$ due to $Q_1 \geq 0$. By writing monomials in $P_1 Q_1$ as quadratic monomials if possible and using $C_{3,1,2}$ to eliminate non-quadratic monomials, we obtain

$$\begin{align*}
R_1 &= P_1 Q_1 - (\frac{1}{8} q_{1,2} \tilde{R}_1 - q_{1,1} \tilde{R}_1 - \frac{1}{5} q_{1,3} \tilde{R}_1) \\
   &= q_{1,1}(2m_1 m_2 - m_2^3) + q_{1,2}(\frac{4}{5}m_3^2 - m_2^2) + \frac{q_{1,3}}{5} m_2^2.
\end{align*}$$

S5. By writing $E_{2,3,1}$ as a quadratic form in $m_i$, we have

$$\begin{align*}
\tilde{E}_{2,3,1} &= E_{2,3,1} - \frac{3}{4} \tilde{R}_1 - \tilde{R}_1 - \frac{1}{3} \tilde{R}_2 + \frac{1}{6} \tilde{R}_4 \\
   &= \frac{1}{2} m_1^2 - 3m_1 m_2 - \frac{5}{3} m_2^2 + 2m_2^3.
\end{align*}$$

S6. Since $\tilde{E}_{3,1}, \tilde{R}_1, \tilde{R}_2, R_1$ are quadratic forms in $m_i$, we can use the Matlab software in Appendix A of [21] to obtain the following SOS representation

$$\begin{align*}
&\tilde{E}_{2,3,1} = \sum_{i=1}^2 p_i \tilde{R}_i + R_1 + \sum_{i=1}^3 c_i (\sum_{j=1}^3 c_{ij} m_j)^2, \\
&Q_1 \geq 0, \\
&Q_1 \geq 0,
\end{align*}$$

where

$$\begin{align*}
p_1 &= \frac{6}{5}, p_2 = -2, c_1 = \frac{1}{2}, c_{1,1} = 1, c_{1,2} = -3, c_{1,3} = 2, \\
q_{1,1} &= q_{1,2} = q_{1,3} = c_2 = c_3 = 0.
\end{align*}$$

S7. We obtain

$$\begin{align*}
E_{2,3,1} &= \frac{3}{4} R_1 + R_2 + \frac{1}{4} R_3 + \frac{1}{8} R_4 - \frac{7}{4} R_5 - \frac{1}{4} R_6 \\
   &\quad + \sum_{i=1}^3 c_i (\sum_{j=1}^3 c_{ij} m_j)^2.
\end{align*}$$
From Theorem 1 and (23), we have
\[
\frac{d^3 H(X_t)}{dt^3} - \frac{21}{2} \mathbb{E} \left[ \frac{(f_1^2 - f_2)^3}{f^6} \right] = \int \frac{E_{2,3,1}}{p_t} dx_t
\]
\[
= \int \frac{1}{8} \left( \frac{3}{4} R_1 + \frac{1}{4} R_2 + \frac{1}{8} R_3 \right) dx_t
\]
\[
- \frac{7}{4} R_5 - \frac{1}{4} R_6 + \sum_{i=1}^{3} c_i(\sum_{j=i}^{3} c_{i,j} m_j)^2 \right) dx_t
\]
\[
= \int \frac{(m_1 - 3m_2 + 2m_3)^2}{2p_t} \quad \text{dx}_t
\]
\[
\geq 0.
\]

Thus, an explicit and strict proof is given for $C_2(3, 1)$. Note that this example is also considered in [18] by the pure SDP-based method, which is a semi-automatic algorithm. See Table 1 for the time used to provide analytical proof of this example by our automatic method on a computer.

3. Proof of $C_1(3,n)$ for $n = 2, 3, 4$

In this section, we use the procedure in Section 2.2 to prove $C_1(3, n)$ for $n = 2, 3, 4$.

3.1. Compute $E_{1,3,n}$

In step 1, we compute $E_{1,3,n}$ in (8) and (20):

\[
\frac{1}{2} \frac{d^2}{dt^2} \left( \int_{\mathbb{R}^n} \frac{1}{p_t} \left\| \nabla p_t \right\|^2 dx_t \right) \leq \int_{\mathbb{R}^n} \frac{E_{1,3,n}}{p_t^2} dx_t,
\]

where

\[
E_{1,3,n} = \sum_{a=1}^{n} \sum_{b=1}^{n} \sum_{c=1}^{n} F_{3,a,b,c}
\]

and

\[
F_{3,a,b,c} = \frac{p_t^4}{4} \frac{\partial^3 p_t}{\partial x_{a,i} \partial x_{b,j} \partial x_{c,k}} - \frac{p_t^1}{4} \frac{\partial p_t}{\partial x_{a,i} \partial x_{b,j} \partial x_{c,k}} \frac{\partial^3 p_t}{\partial x_{a,i} \partial x_{b,j} \partial x_{c,k}} - \frac{p_t^3}{2} \frac{\partial^3 p_t}{\partial x_{a,i} \partial x_{b,j} \partial x_{c,k}} \frac{\partial^2 p_t}{\partial x_{a,i} \partial x_{b,j} \partial x_{c,k}}.
\]

3.2. Compute the Third-Order Constraints

In step 2, we obtain the third-order constraints. We introduce the notation

\[
\mathcal{V}_{a,b,c} = \left\{ \frac{\partial^h p_t}{\partial x_{a,i} \partial x_{b,j} \partial x_{c,k}} : h = h_1 + h_2 + h_3 \in [5]_0 \right\},
\]

where $a, b, c$ are variables taking values in $[n]$. Then,

\[
\mathcal{P}_{3,n} = \bigcup_{a=1}^{n} \bigcup_{b=1}^{n} \bigcup_{c=1}^{n} \mathcal{V}_{a,b,c}.
\]

The third-order integral constraints are:

\[
C_{3,n} = \{ R_{3,a,b,c}^{(3)} : i = 1, \ldots, 955; a,b,c \in [n] \},
\]
R_{i,a,b,c}^{(3)} in the form of lengthy formulas can be found in [23]. Note that we do not use all the third-order constraints in [23].

3.3. Proof of C_{1}(3,2)

The proof follows Procedure 2 with E_{1,3,2} given in (26) as the input. To make the proof explicit, we will give the key expressions.

In Step S1, the new variables are M_{3,2} and are listed in the lexicographical monomial order:

\[ m_1 = p_l^3 \frac{\partial p_l}{\partial x_{2,l}}, m_2 = p_l^3 \frac{\partial^2 p_l}{\partial x_1,l \partial x_{2,l}}, \]
\[ m_3 = p_l^3 \frac{\partial^3 p_l}{\partial x_1,l^3}, m_4 = p_l^3 \frac{\partial^3 p_l}{\partial x_1,l^4}, \]
\[ m_5 = p_l^3 \frac{\partial^3 p_l}{\partial x_1,l^3}, m_6 = p_l^3 \frac{\partial^3 p_l}{\partial x_2,l^3}, \]
\[ m_7 = p_l^3 \frac{\partial^3 p_l}{\partial x_1,l^3}, m_8 = p_l^3 \frac{\partial^3 p_l}{\partial x_1,l^3}, \]
\[ m_9 = p_l^3 \frac{\partial^3 p_l}{\partial x_1,l^2}, m_{10} = p_l^3 \frac{\partial^3 p_l}{\partial x_1,l^2}, \]
\[ m_{11} = \left( \frac{\partial p_l}{\partial x_{2,l}} \right)^3, m_{12} = \left( \frac{\partial p_l}{\partial x_{2,l}} \right)^2 \frac{\partial p_l}{\partial x_{1,l}}, \]
\[ m_{13} = \frac{\partial p_l}{\partial x_{2,l}} \left( \frac{\partial p_l}{\partial x_{1,l}} \right)^2, m_{14} = \left( \frac{\partial p_l}{\partial x_{1,l}} \right)^3. \]

In Step S2, the constraints are

\[ C_{3,2} = \{ R_{i,a,b,c}^{(3)} : j = 1, \ldots, 955; a, b, c \in [2]\}. \]

Removing the repeated ones, we have N_1 = 135. We obtain C_{3,2,1} and C_{3,2,2}, which contain 48 and 52 constraints, respectively.

In Step S3, there exist 15 intrinsic constraints:

\[ m_5 m_8 = m_6 m_9, m_5 m_{10} = m_6 m_{11}, \]
\[ m_5 m_{13} = m_6 m_{12}, m_5 m_{14} = m_6 m_{13}, m_7 m_{10} = m_8 m_9, \]
\[ m_7 m_{12} = m_8 m_{11}, m_7 m_{13} = m_8 m_{12}, m_7 m_{14} = m_8 m_{13}, \]
\[ m_{10} m_{12} = m_{10} m_{11}, m_{10} m_{13} = m_{10} m_{12}, m_{10} m_{14} = m_{10} m_{13}, \]
\[ m_{11} m_{13} = m_{12}^2, m_{11} m_{14} = m_{12} m_{13}, m_{12} m_{14} = m_{13}^2. \]

Thus, \( \tilde{C}_{3,2,1} \) contains 63 constraints and \( N_3 = 63 \).

Step S4 is not needed in the proof of this case.

In Step S5, by eliminating the non-quadratic monomials in \( E_{1,3,2} \) using \( C_{3,2,2} \) to obtain a quadratic form in \( m_i \) and then simplifying the quadratic form using \( C_{3,2,1} \), we have
\[ \hat{E}_{1,3,2} = E_{1,3,2} - \left( \frac{3}{4} \hat{R}_{17} - \frac{1}{6} \hat{R}_{12} - \frac{1}{6} \hat{R}_{13} + \frac{7}{6} \hat{R}_{18} - \frac{1}{2} \hat{R}_{32} \right) \]
\[ - \frac{1}{2} \hat{R}_{34} - \frac{5}{8} \hat{R}_{35} - \frac{1}{2} \hat{R}_{40} - \frac{1}{12} \hat{R}_{2} - \frac{1}{8} \hat{R}_{5} - \frac{1}{4} \hat{R}_{6} \]
\[ + \frac{1}{2} \hat{R}_{7} + \frac{1}{4} \hat{R}_{8} + \frac{1}{2} \hat{R}_{18} + \frac{1}{4} \hat{R}_{19} - \frac{1}{8} \hat{R}_{39} - \frac{1}{4} \hat{R}_{46} \]
\[ + \frac{1}{2} \hat{R}_{48} - \frac{1}{8} \hat{R}_{49} + \frac{1}{4} \hat{R}_{53} \]
\[ = \frac{1}{2} \bar{m}_{1}^{2} - m_{1} m_{5} + \frac{3}{2} \bar{m}_{2}^{2} - 3 m_{2} m_{6} + \frac{3}{2} \bar{m}_{3}^{2} + \frac{1}{2} \bar{m}_{4}^{2} \]
\[ - 2 m_{4} m_{6} - m_{4} m_{7} - m_{4} m_{10} - \frac{1}{2} m_{2}^{2} + \frac{3}{2} \bar{m}_{8}^{2} - 3 \bar{m}_{7}^{2} \]
\[ - 2 m_{7} m_{10} + 3 m_{8} - \frac{5}{2} \bar{m}_{9}^{2} - \frac{3}{2} m_{9} m_{11} + 21 m_{9} m_{13} \]
\[ - \frac{1}{2} m_{10}^{2} + \frac{3}{5} \bar{m}_{11}^{2} + 3 \bar{m}_{12} - \frac{15}{2} m_{13}^{2} + \frac{3}{5} \bar{m}_{14}^{2} \]

In Step S6, using the Matlab program in [23] with \( \hat{E}_{1,3,2} \) and \( \hat{C}_{3,2,1} \) as the input, we find an SOS representation for \( \hat{E}_{1,3,2} \). Thus, by Theorem 1, \( C_{1}(3,2) \) is strictly proved.

### 3.4. Proof of \( C_{1}(3,3) \)

The proof follows Procedure 2 with \( E_{1,3,3} \) given in (29) as the input. The detailed lengthy formulas can be seen in [23].

In Step S1, the new variables are \( M_{3,3} = \{ m_{i}, i = 1, \ldots, 38 \} \) which is the set of all monomials in \( \mathbb{R}[P_{3,3}] \) with a degree of 3 and a total order of 3, and which are listed in the lexicographical monomial order.

In Step S2, the constraints are: \( C_{3,3} = \{ R_{i,j,k}^{(3)} : i = 1, \ldots, 955 \} \). \( N_{1} = 955 \). We obtain \( C_{3,3,1} \) and \( C_{3,3,2} \), which contain 350 and 328 constraints, respectively.

In Step S3, there exist 189 intrinsic constraints. In total, \( \hat{C}_{3,3,1} \) contains 539 constraints. Using \( \mathbb{R} \)-Gaussian elimination in \( \text{Span}_{\mathbb{R}}(\hat{C}_{3,3,1}) \) shows that 512 of these 539 constraints are linearly independent, so \( N_{3} = 512 \).

Step S4 is not needed in the proof of this case.

In Step S5, by eliminating the non-quadratic monomials in \( E_{1,3,3} \) using \( C_{3,3,2} \) and then simplifying the expression using \( C_{3,3,1} \), we obtain \( \hat{E}_{1,3,3} \) written as a quadratic form in \( m_{i} \).

In Step S6, using the Matlab program in [23] with \( \hat{E}_{1,3,3} \) and \( \hat{C}_{3,3,1} \) as the input, we find an SOS representation for \( \hat{E}_{1,3,3} \). Thus, using Theorem 1, \( C_{1}(3,3) \) is strictly proved.

### 3.5. Proof of \( C_{1}(3,4) \)

The proof follows Procedure 2 with \( E_{1,3,4} \) given in (29) as the input. The detailed lengthy formulas can be seen in [23].

In Step S1, the new variables are \( M_{3,4} = \{ m_{i}, i = 1, \ldots, 80 \} \) which is the set of all monomials in \( \mathbb{R}[P_{3,4}] \) with a degree of 3 and a total order of 3, and which are listed in the lexicographical monomial order.

In Step S2, we obtain \( C_{3,4} = \{ R_{i,j,k}^{(3)}, R_{i}^{(0)}, R_{k}^{(2)} : i = 1, \ldots, 955, j = 1, \ldots, 8, k = 1, \ldots, 20, a, b, c \in \{ 4 \} \} \). Removing the repeated ones, we have \( N_{1} = 3172 \). We obtain \( C_{3,4,1} \) and \( C_{3,4,2} \) which contain 1120 and 975 constraints, respectively.

In Step S3, there exist 1080 intrinsic constraints. In total, \( \hat{C}_{3,4,1} \) contains 2200 constraints. Only 1966 constraints in \( \hat{C}_{3,4,1} \) are \( \mathbb{R} \)-linearly independent, so \( N_{2} = 1966 \).

Step S4 is not needed in the proof of this case.

In Step S5, by eliminating the non-quadratic monomials in \( E_{1,3,4} \) using \( C_{3,4,2} \) to obtain a quadratic form in \( m_{i} \) and then simplifying the quadratic form with \( C_{3,4,1} \), we obtain \( \hat{E}_{1,3,4} \) which is written as a quadratic form in \( m_{i} \).

In Step S6, using the Matlab program in [23] with \( \hat{E}_{1,3,4} \) and \( \hat{C}_{3,4,1} \) as the input, we find an SOS representation for \( \hat{E}_{1,3,4} \). Thus, using Theorem 1, \( C_{1}(3,4) \) is strictly proved.
4. Proof of $C_3(3,n)$ for $n = 2, 3, 4$ under the Log-Concave Condition

In this section, we use the procedure in Section 2.2 to prove $C_3(3,n)$ for $n = 2, 3, 4$ under the log-concave condition. The detailed lengthy formulas can be seen in [21].

4.1. Compute $E_{3,3,n}$

In step 1, we compute $E_{3,3,n}$ in (8) and (20):

$$\frac{1}{2} \frac{d^2}{dt^2} \left( \frac{\| \nabla p_t \|^2}{p_t} \right) - \frac{1}{n^3} \mathbb{E} \left( \frac{\| \nabla p_t \|^2 - p_t \nabla^2 p_t}{p_t^2} \right)^3 \tag{29}$$

where

$$E_{3,3,n} = \sum_{a=1}^{n} \sum_{b=1}^{n} \sum_{c=1}^{n} E_{3,a,b,c}$$

and

$$E_{3,a,b,c} = \frac{p_t^4}{4} \frac{\partial^3 p_t}{\partial x_{a,l} \partial x_{b,l} \partial x_{c,l}} \frac{\partial^3 p_t}{\partial x_{a,l} \partial x_{b,l} \partial x_{c,l}} - \frac{p_t^3}{4} \frac{\partial^3 p_t}{\partial x_{a,l} \partial x_{b,l} \partial x_{c,l}} \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{b,l} \partial x_{c,l}} - \frac{p_t^3}{4} \frac{\partial^3 p_t}{\partial x_{a,l} \partial x_{b,l} \partial x_{c,l}} \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{b,l} \partial x_{c,l}} + \frac{p_t^2}{4} \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{b,l}} \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{b,l}} - \frac{p_t^2}{8} \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{b,l}} \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{b,l}} - \frac{1}{n^3} \left[ \left( \frac{\partial p_t}{\partial x_{a,l}} \right)^2 - p_t \left( \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{b,l}} \right) \right] \left[ \left( \frac{\partial p_t}{\partial x_{a,l}} \right)^2 - p_t \left( \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{b,l}} \right) \right].$$

4.2. Compute the Third-Order Log-Concave Constraints

In step 2, we obtain the third-order log-concave constraints.

From Lemma 3, we can compute the third-order log-concave constraints:

$$C_{3,2} = \{ R_1 = -\triangle_{1,1} Q_1, R_2 = -\triangle_{1,2} Q_2, R_3 = \triangle_{2,1} Q_3 \},$$

where $Q_1, Q_2 \in \text{Span}_g(\mathcal{M}_{4,4})$ and $Q_3 \in \text{Span}_g(\mathcal{M}_{2,2})$. Note that $C_{3,2}$ does not contain all the log-concave constraints in Lemma 3. The constraints $C_{3,2}$ are enough for our purpose in this paper.

For $n > 2$, we give certain log-concave constraints in a special form, which are needed in the proof procedure in Section 4.3. Let

$$\nabla_1 p_t = \left( \frac{\partial p_t}{\partial x_{a,l}}, \frac{\partial p_t}{\partial x_{a,l}}, \frac{\partial p_t}{\partial x_{a,l}} \right),$$

$$L_1(p_t) = p_t H_1(p_t) - \nabla_1^T p_t \nabla_1 p_t,$$

where

$$H_1(p_t) = \begin{bmatrix} \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{b,l}} & \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{c,l}} & \frac{\partial^2 p_t}{\partial x_{a,l} \partial x_{d,l}} \\ \frac{\partial^2 p_t}{\partial x_{b,l} \partial x_{c,l}} & \frac{\partial^2 p_t}{\partial x_{b,l} \partial x_{d,l}} & \frac{\partial^2 p_t}{\partial x_{b,l} \partial x_{e,l}} \\ \frac{\partial^2 p_t}{\partial x_{c,l} \partial x_{d,l}} & \frac{\partial^2 p_t}{\partial x_{c,l} \partial x_{e,l}} & \frac{\partial^2 p_t}{\partial x_{c,l} \partial x_{f,l}} \end{bmatrix},$$

and $\triangle_{l,j}^i, i = 1, \ldots, L_k$ the kth-order principle minors of $L_1(p_t)$. Let $\mathcal{M}_k'$ be the set of all monomials in $V_{a,b,c}$ (defined in (27)) which have a degree of $k$ and a total order of $k$. We have

$$C_{3,n} = \{ -\triangle_{1,1} Q_{1,1}, -\triangle_{1,2} Q_{1,2}, -\triangle_{1,3} Q_{1,3}, \triangle_{1,1} Q_{2,1}, \triangle_{1,2} Q_{2,2}, \triangle_{1,3} Q_{2,3}, -\triangle_{1,1} Q_{3,1} \}$$

(31)
where \( Q_{1,i} \in \text{Span}_R(M'_4), \) \( Q_{2,i} \in \text{Span}_R(M'_2), \) and \( Q_{3,1} \in \mathbb{R}. \)

4.3. Proof of \( C_{3}(3,2) \)

The proof follows Procedure 2 with \( E_{3,3,2} \) given in (29) and the constraints in (28) and (30) as the input.

Steps S1–S3 are the same with the proof of the case \( C_1(3,2). \)

In Step S4, we obtain \( \hat{C}(3,2) \) which contains three quadratic-form constraints.

In Step S5, by eliminating the non-quadratic monomials in \( E_{3,3,2} \) using \( C_{3,2,2} \) to obtain a quadratic form in \( m_i \) and then simplifying the quadratic form using \( C_{3,2,1} \), we have

\[
\hat{E}_{3,3,2} = \frac{31}{40} m_{14}^2 - \frac{147}{8} m_{13}^2 - \frac{5}{2} m_7 m_{10} + \frac{15}{4} m_5^2 - \frac{25}{8} m_3^2
\]

\[
- \frac{31}{16} m_{9} m_{11} + \frac{207}{8} m_9 m_{13} - \frac{5}{8} m_7^2 + \frac{1}{2} m_1^2
\]

\[
- \frac{5}{4} m_4 m_5 + \frac{31}{40} m_{12}^2 + \frac{31}{8} m_5^2 + \frac{1}{2} m_4^2 - \frac{5}{2} m_4 m_6
\]

\[
- \frac{5}{4} m_4 m_7 + \frac{3}{2} m_5^2 - \frac{15}{4} m_7^2 - \frac{5}{4} m_4 m_{10}
\]

\[
- \frac{5}{8} m_3^2 + \frac{15}{8} m_6^2 + \frac{3}{2} m_2^2 - \frac{15}{4} m_2 m_6.
\]

In Step S6, using the Matlab software in Appendix A [21] with \( \hat{E}_{3,3,2}, \hat{C}_{3,2,1} \) and \( \hat{C}_{3,2} \) as the input, we find an SOS representation for \( \hat{E}_{3,3,2} \). Thus, \( C_{3}(3,2) \) is proved under the log-concave condition. The Maple program for proving \( C_{3}(3,2) \) can be found at https://github.com/cmyuanmmrc/codeforepi/ (accessed on 15 July 2020).

Remark 4. We fail to prove \( C_{2}(3,2) \) even under the log-concave condition using the above procedure. Specifically, we cannot find an SOS representation for \( \hat{E}_{2,3,2} \) in Step S6. Since the SDP algorithm is not complete for problem (21), we cannot say that an SOS representation does not exist for \( \hat{E}_{2,3,2} \). The Maple program for \( C_{2}(3,2) \) can be found at https://github.com/cmyuanmmrc/codeforepi/ (accessed on 15 July 2020).

4.4. Proof of \( C_{3}(3,3) \) and \( C_{3}(3,4) \)

In this subsection, we prove \( C_{3}(3,3), C_{3}(3,4) \). Motivated by symmetric functions, for any function \( f(a,b,c) \), we have

\[
\sum_{a,b,c=1}^{n} f(a,b,c) = \sum_{1 \leq a < b < c}^{n} \left\{ \frac{2}{(n - 1)(n - 2)} \left[ f(a,a,a) + f(b,b,b) + f(c,c,c) \right] + \frac{1}{n-2} \left[ f(a,a,b) + f(a,b,a) + f(b,a,a) + f(a,a,c) + f(a,c,a) + f(c,a,a) + f(b,b,a) + f(b,a,b) + f(a,b,b) + f(b,b,c) + f(b,c,b) + f(c,b,b) + f(c,c,a) + f(c,a,c) + f(c,c,b) + f(c,b,c) + f(b,c,c) + f(b,c,b) + f(b,b,c) \right] \right\}.
\]

From (29) and (32), we obtain

\[
E_{3,3,n} = \sum_{a=1}^{n} \sum_{b=1}^{n} \sum_{c=1}^{n} E_{3,a,b,c} = \sum_{1 \leq a < b < c \leq n}^{n} f_{3,3,n}.
\]
5. Proof of C3

In the above procedure, in the same way, C3 is strictly proved. The Maple program used to prove C3 as the input, we find an SOS representation for m, using R as the differential operators without giving concrete values to a, b, and c.

First, we prove C3 using Procedure 2 with J3,3 given in (33) and the constraints in (28) and (31) as the input.

In Step S1, the new variables are M' = \{m_i, i = 1, \ldots, 38\}, which is the set of all the monomials in R[V_{a,b,c}] with a degree of 3 and a total order of 3.

In Step S2, the constraints are: C3,n = \{R_{i,a,b,c} : i = 1, \ldots, 955\}, N_1 = 955. We obtain C_{3,n,1} and C_{3,n,2}, which contain 350 and 328 constraints, respectively.

In Step S3, there exist 189 intrinsic constraints. In total, C_{3,n,1} contains 539 constraints. Using R-Gaussian elimination in Span_R(\hat{C}_{3,n,1}) shows that 512 of these 539 constraints are linearly independent, thus N_0 = 512.

In Step S4, we obtain \hat{C}_{3,n} from C_{3,n}, which contains six constraints.

In Step S5, eliminating the non-quadratic monomials in j3,3,3 using C_{3,n,2} and then simplifying the expression using C_{3,n,1}, we obtain j3,3,3, which is written as a quadratic form in m_i.

In Step S6, using the Matlab software in Appendix A [21] with \hat{j}_{3,3,3}, \hat{C}_{3,n,1} and \hat{C}_{3,n} as the input, we find an SOS representation for j3,3,3. Thus, using Theorem 1, C3(3,3) is strictly proved. The Maple program used to prove C3(3,3) can be found at https://github.com/cmyuanmmrc/codeforepi/ (accessed on 15 July 2020).

To prove C3(3,4), we just need to replace the input from j3,3,3 with j3,3,4 in Step S5 in the above procedure. In the same way, C3(3,4) can be strictly proved. The Maple program used to prove C3(3,4) can be found at https://github.com/cmyuanmmrc/codeforepi/ (accessed on 15 July 2020).

5. Proof of C3(4,2)

In this section, we use the procedure in Section 2.2 to prove C3(4,2) under the log-concave condition.

In Step 1, we compute E_{3,A,n} in (8) and (20):

\[
\frac{1}{2} \int_{\mathbb{R}^n} \left( \frac{\|\nabla p_t\|^2}{p_t} \right) - \frac{3}{n^4} \mathbb{E} \left( \frac{\|\nabla p_t\|^2 - p_t \nabla^2 p_t}{p_t^2} \right)^4 \, dx_t
\]

where E_{3,A,n} = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \sum_{l=1}^{n} E_{4,a,b,c,d}. For brevity, we omit the concrete expression of E_{4,a,b,c,d}.

In Step 2, based on Lemma 2, we obtain 589 fourth-order constraints:

\[
C_{4,2} = \{ R_i^{(2)} : i = 1, \ldots, 589 \} \subset \mathbb{R} | P_{4,2} | \text{ and } N_1 = 589.
\]

From (33), if we prove J3,3,n ≥ 0, then E3,3,n ≥ 0. It is clear that J3,3,n has many fewer terms than E3,3,n.

In J3,3,n given in (33) and the constraints in (28) and (31), we may consider \frac{\partial}{\partial x_i} and \frac{\partial}{\partial y_j} as the differential operators without giving concrete values to a, b, and c.

From (33), if we prove \hat{C}_j, there exist 189 intrinsic constraints. In total, \hat{C}_{3,n,1} contains 539 constraints. Using R-Gaussian elimination in Span_R(\hat{C}_{3,n,1}) shows that 512 of these 539 constraints are linearly independent, thus N_0 = 512.

In Step S4, we obtain \hat{C}_{3,n} from C_{3,n}, which contains six constraints.

In Step S5, eliminating the non-quadratic monomials in j3,3,3 using C_{3,n,2} and then simplifying the expression using C_{3,n,1}, we obtain j3,3,3, which is written as a quadratic form in m_i.

In Step S6, using the Matlab software in Appendix A [21] with \hat{j}_{3,3,3}, \hat{C}_{3,n,1} and \hat{C}_{3,n} as the input, we find an SOS representation for j3,3,3. Thus, using Theorem 1, C3(3,3) is strictly proved. The Maple program used to prove C3(3,3) can be found at https://github.com/cmyuanmmrc/codeforepi/ (accessed on 15 July 2020).

To prove C3(3,4), we just need to replace the input from j3,3,3 with j3,3,4 in Step S5 in the above procedure. In the same way, C3(3,4) can be strictly proved. The Maple program used to prove C3(3,4) can be found at https://github.com/cmyuanmmrc/codeforepi/ (accessed on 15 July 2020).

5. Proof of C3(4,2)

In this section, we use the procedure in Section 2.2 to prove C3(4,2) under the log-concave condition.

In Step 1, we compute E_{3,A,n} in (8) and (20):

\[
\frac{1}{2} \int_{\mathbb{R}^n} \left( \frac{\|\nabla p_t\|^2}{p_t} \right) - \frac{3}{n^4} \mathbb{E} \left( \frac{\|\nabla p_t\|^2 - p_t \nabla^2 p_t}{p_t^2} \right)^4 \, dx_t
\]

where E_{3,A,n} = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \sum_{l=1}^{n} E_{4,a,b,c,d}. For brevity, we omit the concrete expression of E_{4,a,b,c,d}.

In Step 2, based on Lemma 2, we obtain 589 fourth-order constraints:

\[
C_{4,2} = \{ R_i^{(2)} : i = 1, \ldots, 589 \} \subset \mathbb{R} | P_{4,2} | \text{ and } N_1 = 589.
\]
Using Lemma 3, we obtain three fourth-order log-concave constraints:

\[ C_{4,2} = \{ -\triangle_{1,1}Q_{1,1}, -\triangle_{1,2}Q_{1,2}, \triangle_{2,1}Q_{2,1} \} \]

where \( Q_{1,1}, Q_{1,2} \in \text{Span}_{\mathbb{R}}(M_{6,2}) \) and \( Q_{2,1} \in \text{Span}_{\mathbb{R}}(M_{4,2}) \).

In step 3, we use Procedure 2 to compute the SOS representations (13) and (14) with \( E_{3,4,n}, C_{4,2} \), and \( C_{4,2} \) as the input.

In Step S1, the new variables are \( M_{4,2} = \{ m_i, i = 1, \ldots, 33 \} \), which is the set of all monomials in \( \mathbb{R}[P_{4,2}] \) with a degree of 4 and a total order of 4, and which is listed in the lexicographical monomial order.

In Step S2, using Gaussian elimination for \( C_{4,2} = \{ R^{(2)}_i : i = 1, \ldots, 589 \} \), we obtain \( C_{4,2,1} \) and \( C_{4,2,2} \), which contain 266 and 182 constraints, respectively.

In Step S3, there exist 182 intrinsic constraints. Thus, \( \hat{C}_{4,2,1} \) contains 448 constraints. Using \( \mathbb{R} \)-Gaussian elimination in \( \text{Span}_{\mathbb{R}}(\hat{C}_{4,2,1}) \) shows that 417 of these 448 constraints are linearly independent, so \( N_3 = 417 \).

In Step S4, we obtain \( \hat{C}(4,2) \), which contain three log-concave constraints, so \( N_2 = 3 \).

In Step S5, by eliminating the non-quadratic monomials in \( E_{3,4,2} \) using \( \hat{C}_{4,2,2} \) to obtain a quadratic form in \( m_i \) and then simplifying the quadratic form using \( C_{4,2,1} \), we obtain \( \hat{E}_{3,4,2} \) which is written as a quadratic form in \( m_i \).

In Step S6, using the Matlab software in Appendix A of [21] with \( \hat{E}_{3,4,2}, \hat{C}_{4,2,1} \), and \( \hat{C}(4,2) \) as the input, we find an SOS representation for \( \hat{E}_{3,4,2} \). Thus, using Theorem 1, \( C_3(4,2) \) is strictly proved under the log-concave condition. The Maple program used to prove \( C_3(4,2) \) can be found at https://github.com/cmyuanmmrc/codeforepi/ (accessed on 15 July 2020).

6. Conclusions

In this paper, three conjectures \( C_l(m,n) \) for \( l = 1, 2, 3 \) concerning the lower bound for the derivatives of \( H(X_l) \) are considered. We propose a general procedure to prove inequities similar to \( C_l(m,n) \). We first consider one of the conjectures of McKean \( C_1(m,n) : \) \(-1\)^{m+1} (\frac{d^m}{dt^m}) H(X_l) \geq 0 \) in the multivariate case, and prove \( C_1(3,2), C_1(3,3) \) and \( C_1(3,4) \). This conjecture is also mentioned in Villani’s paper [14], and is named the super-H theorem. Motivated by \( C_2(m,n) \), we further propose the following weaker conjecture \( C_3(m,n) : \) \(-1\)^{m+1} (\frac{d^m}{dt^m}) H(X_l) \geq (1-m)\frac{d^m}{dt^m} H(X_{Gl}) \). Using our procedure, we prove \( C_3(3,2), C_3(3,3), C_3(3,4) \) and \( C_3(4,2) \) under the log-concave condition.

In the univariate case \( (n = 1) \), \( C_1(3,1) \) and \( C_1(4,1) \) were proved [16] and \( C_1(5,1) \) cannot be proved with the SDP approach (In this paper, when we say \( C_3(m,n) \) cannot be proved with the SDP approach, we mean that the software in Appendix A of [21] terminates and gives a negative answer for problem (21)) [18,22]. \( C_2(3,1), C_2(4,1) \), and \( C_2(5,1) \) were proved under the log-concave condition [18]. We try to prove \( C_2(6,1) \) under the log-concave condition. However, due to the accuracy of the SDP software, we cannot find an explicit SOS representation. In the multivariate case, \( C_1(3,2), C_1(3,3), \) and \( C_1(3,4) \) were proved and \( C_1(4,2) \) cannot be proved with the SDP approach [22]. For \( C_1(3,n), n > 4 \), the corresponding SDP problem is too large for the Matlab software in Appendix A [23].

In this paper, \( C_3(3,2), C_3(3,3), C_3(3,4), \) and \( C_3(4,2) \) were proved under the log-concave condition, and \( C_2(3,2), C_2(3,3), C_2(3,4), \) and \( C_2(4,2) \) cannot be proved with the SDP approach under the log-concave condition. For \( C_3(3,n), n > 4 \) and \( C_3(4,n), n > 2 \), the corresponding SDP problems are too large for the Matlab software in Appendix A [21].

In order to use the SDP approach to prove more difficult problems, two kinds of improvements are needed. First, it is easy to see that the size of \( E_l(m,n) \) and the numbers of the constraints increase exponentially as \( m \) and \( n \) become larger. Thus, we need to find certain rules which could be used to simplify the computation to solve problems such as \( C_1(3,n)(n > 4) \) and \( C_3(3,n)(n > 4) \) under the log-concave condition. Second, in many cases, such as \( C_1(5,1) \) and \( C_2(3,2) \) under the log-concave constraint, the SDP software terminates and gives a negative answer. Since the SDP method is not complete for our
problem, we do not know whether an SOS representation exists. We thus need a complete method to solve problem (13). Another problem is to find more constraints besides those used in this paper in order to increase the power of the approach.

**Author Contributions:** Conceptualization, L.G.; formal analysis, L.G., C.-M.Y. and X.-S.G.; funding acquisition, L.G., C.-M.Y. and X.-S.G.; investigation, L.G. and X.-S.G.; methodology, L.G.; project administration, X.-S.G.; resources, L.G.; software, L.G.; supervision, X.-S.G. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work is supported by NSFC 11688101 and NKRDP 2018YFA0704705, Beijing Natural Science Foundation (No. Z190004), and the Fundamental Research Funds for the Central Universities 2021NTST32.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The authors would like to thank the reviewers for their invaluable comments.

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**

1. Shannon, C.E. A mathematical theory of communications. *Bell Syst. Tech. J.* 1948, 27, 379–423. [CrossRef]
2. Stam, A.J. Some inequalities satisfied by the quantities of information of Fisher and Shannon. *Inf. Control* 1959, 2, 101–112. [CrossRef]
3. Blachman, N.M. The convolution inequality for entropy powers. *IEEE Trans. Inf. Theory* 1965, 11, 267–271. [CrossRef]
4. Lieb, E.H. Proof of an entropy conjecture of Wehrl. *Commun. Math. Phys.* 1978, 62, 35–41. [CrossRef]
5. Verdú, S.; Guo, D. A simple proof of the entropy-power inequality. *IEEE Trans. Inf. Theory* 2006, 52, 2165–2166. [CrossRef]
6. Rioul, O. Information theoretic proofs of entropy power inequalities. *IEEE Trans. Inf. Theory* 2011, 57, 33–55. [CrossRef]
7. Bergmans, P. A simple converse for broadcast channels with additive white Gaussian noise. *IEEE Trans. Info. Theory* 1974, 20, 279–280. [CrossRef]
8. Zamir, R.; Feder, M. A generalization of the entropy power with respect to added Gaussian noise. *IEEE Trans. Inf. Theory* 1993, 39, 1723–1728. [CrossRef]
9. Liu, T.; Viswanath, P. An extremal inequality motivated by multiterminal information-theoretic problems. *IEEE Trans. Inf. Theory* 2007, 53, 1839–1851. [CrossRef]
10. Wang, L.; Madiman, M. A new approach to the entropy power inequality, via rearrangements. In Proceedings of the 2013 IEEE International Symposium on Information Theory, Istanbul, Turkey, 7–12 July 2013; pp. 599–603.
11. Guo, L.; Yuan, C.M.; Gao, X.S. A Generalization of the Concavity of Rényi Entropy Power. *Entropy* 2021, 23, 1593. [CrossRef] [PubMed]
12. Costa, M.H.M. A new entropy power inequality. *IEEE Trans. Inf. Theory* 1985, 31, 751–760. [CrossRef]
13. Dembo, A. Simple proof of the concavity of the entropy power with respect to added Gaussian noise. *IEEE Trans. Inf. Theory* 1989, 35, 887–888. [CrossRef]
14. Villani, C. A short proof of the concavity of entropy power. *IEEE Trans. Inf. Theory* 2000, 46, 1695–1696. [CrossRef]
15. Toscani, G. A concavity property for the reciprocal of Fisher information and its consequences on Costa’s EPI. *Phys. A Stat. Mech. Appl.* 2015, 432, 352. [CrossRef]
16. Cheng, F.; Geng, Y. Higher order derivatives in Costa’s entropy power inequality. *IEEE Trans. Inf. Theory* 2015, 61, 5892–5905. [CrossRef]
17. McKean, H.P., Jr. Speed of approach to equilibrium for Kac’s caricature of a Maxwellian gas. *Arch. Ration. Mech. Anal.* 1966, 21, 343–367. [CrossRef]
18. Zhang, X.; Anantharam, V.; Geng, Y. Gaussian optimality for derivatives of differential entropy using linear matrix inequalities. *Entropy* 2018, 20, 182. [CrossRef] [PubMed]
19. Boyd, S.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press: Cambridge, MA, USA, 2004.
20. Vandenberghe, L.; Boyd, S. Semidefinite programming. *SIAM Rev.* 1996, 38, 49–95. [CrossRef]
21. Guo, L.; Yuan, C.M.; Gao, X.S. Lower bound on derivatives of Costa’s differential entropy. *arXiv* 2020, arXiv:2007.10145.
22. Guo, L.; Yuan, C.M.; Gao, X.S. Lower bound for derivatives of Costa’s differential entropy. In Proceedings of the IEEE International Symposium on Information Theory (ISIT), Melbourne, Australia, 12–20 July 2021; pp. 1516–1521.
23. Guo, L.; Yuan, C.M.; Gao, X.S. Prove Costa’s entropy power inequality and high order inequality for differential entropy with semidefinite programming. *arXiv* 2020, arXiv:2004.08543.