Multiterminal Source Coding with Copula Regression for Wireless Sensor Networks Gathering Diverse Data

Evaneglos Zimos, Student Member, IEEE, Dimitris Toupakaris, Member, IEEE, Adrian Munteanu, Member, IEEE, and Nikos Deligiannis, Member, IEEE

Abstract—Efficient data compression at a low processing and communication cost is a key challenge in wireless sensor networks. In this paper, we propose a novel multiterminal source code design, which, contrary to prior work, utilizes both the intra- and the inter-sensor data dependencies. The former is exploited by applying simple DPCM followed by arithmetic entropy coding at each distributed encoder. This approach limits the encoding complexity and provides for a flexible design that adapts to variations in the number of operating sensors. Moreover, we propose a regression method applied at the joint decoder, which aims at leveraging the inter-sensor data dependencies. Unlike existing work that focuses on homogeneous data types, the proposed method makes use of copula functions, namely, a statistical model that captures the dependence structure amongst heterogeneous data types. Experimentation using real sensor measurements—taken from the Intel-Berkeley database—shows that the proposed system achieves significant improvement compared to state-of-the-art multiterminal and distributed source coding schemes.

Index Terms—Wireless sensor networks (WSNs), multiterminal (MT) source coding, distributed source coding (DSC), copula regression, differential pulse-code modulation (DPCM).

I. INTRODUCTION

Wireless sensor networks (WSNs) play a key role in a plethora of emerging applications in a wide span of disciplines, ranging from smart cities and smart water systems to smart metering, home automation and smart farming and agriculture [1]. The vision of such systems is realized by the emerging Internet of Things (IoT), which is supported by the integration of WSNs in the generic internet infrastructure via the 6LoWPAN/IPv6 standard [2]. In the context of these applications, wireless sensors collect diverse correlated information such as light, pressure, temperature, or humidity data, process it, and then transmit it to central nodes for storage and/or further processing [1].

Since wireless sensors are typically powered by batteries that cannot easily be changed or recharged, the primary constraint in the design of WSNs is energy consumption. Power savings can be achieved by reducing the radio emission of the sensors, which, therefore, calls for efficient compression of the transmitted data. To this end, the intra- and inter-sensor data dependencies need to be effectively leveraged without increasing the computational effort at the sensor nodes and without requiring inter-sensor communication. The encoding complexity at the sensor nodes should be as low as possible and the computational burden should be shifted towards energy-robust central nodes (e.g., base stations, fusion centers). Moreover, the encoding design needs to be flexible in terms of rate allocation so as to avoid continuous reconfiguration.

Existing works [3]–[5] propose conventional compression algorithms for WSNs, where low-memory differential pulse-code modulation (DPCM) [6] followed by entropy coding is used. Such compression schemes exploit the intra-sensor data dependencies, namely, the dependencies among consecutive samples collected by each sensor. In order to leverage the dependence among data collected by different nodes, the conventional predictive coding paradigm requires that data be exchanged between the nodes, which in turn implies that inter-sensor communication is established. However, this encoding strategy introduces additional radio transmission requirements for the sensors, thereby leading to rapid battery depletion.

An alternative strategy for efficient data compression in WSNs adheres to distributed source coding (DSC), a paradigm that leverages inter-sensor data dependencies at the decoder side. DSC was initiated by Slepian and Wolf [7], who showed that by separate encoding, two correlated sources can be compressed to their joint entropy with vanishing decoding error probability as the code length goes to infinity. Later, Wyner and Ziv [8] established the rate-distortion bound for lossy compression with decoder side information. They showed that when the source and the side information are jointly Gaussian and the mean-squared error (MSE) is used as the distortion metric, there is no performance loss incurred by not using the side information at the encoder. Recently, this no-rate-loss property has been extended to the case where the source and the side information are binary and correlated by means of the Z-channel [9].

Berger [10] and Tung [11] introduced the multiterminal (MT) source coding problem, which refers to separate lossy encoding and joint decoding of two (or more) correlated sources. From a theoretical perspective, the problem is shown to be challenging: an achievable rate region for the general MT problem is still unknown, but inner and outer bounds have been devised [10], [11]. Theoretical studies have focused on...
special cases such as the quadratic Gaussian, where Gaussian sources and a quadractic distortion criterion are assumed [12].

Towards practical implementations of DSC for WSNs, a two-sensor Slepian-Wolf (SW) coding scheme for temperature monitoring was deployed in [13], where rate adaptation was achieved by means of an entropy tracking algorithm. An alternative SW design using Raptor codes was proposed in [14], [15] for cluster-based WSNs that measure temperature data. Instead of using SW coding as in [13]–[15], the work in [16] devised a Wyner-Ziv (WZ) code construction for WSNs measuring temperature data, which comprised quantization followed by binarization and LDPC encoding. Focusing on the application of wind farm monitoring, an MT code construction [17] was developed to compress wind speed measurements in [18].

Prior studies consider the compression of homogeneous data types such as temperature [13]–[15] or wind speed data [18]. However, many up-to-date applications involve various sensors of heterogeneous modalities measuring diverse yet correlated data (e.g., temperature, humidity, light). In this work, we propose a novel MT source coding scheme that achieves efficient compression by leveraging the dependencies among diverse data types produced by multiple heterogeneous data sources. Our specific contributions are as follows:

• We propose a novel code design for multisensory WSNs, where both intra- and inter-sensor data dependencies are exploited via DPCM and MT source coding, respectively. The proposed system combines the merits of conventional predictive coding [3]–[5]—where only intra-sensor data dependencies are leveraged—and DSC systems [14], [15], [18]–[20]—that only uses inter-sensor dependencies.

• The proposed design is characterized by (i) lightweight encoding, as it applies DPCM to utilize the intra-sensor data dependencies instead of complex vector quantization or trellis-coded quantization (TCQ) as in other works [21]; (ii) optimized compression performance, as it deploys a scalar Lloyd-Max quantizer at each encoder instead of a simple uniform scalar quantizer (USQ) used in other schemes [20]; and (iii) flexibility, since, contrary to classical DSC systems [17], [22]–[27], no system reconfiguration is required when a subset of sensors is not functional.

• Previous work [13], [16], [20] has focused on WSNs collecting homogeneous data and has used a multivariate normal or Laplace distribution to describe the inter-sensor data dependencies. However, in this work, we exploit the data structure among multiple sensors that collect data of different types. In order to accurately express the symmetric and asymmetric dependencies across diverse data sources, we propose the use of statistical models based on copula functions 1 [30], [31]. To this end, we use well-known Elliptical copulas, such as the normal and Student’s t copulas, as well as the Clayton copula, which belongs to the Archimedean copula family. We show that copula functions capture the dependencies among heterogeneous data sources more accurately than the conventional multivariate modeling approach.

The remainder of the paper is organized as follows: Section II gives a brief description of MT source coding without SW compression. Section III presents the proposed coding design, whereas Section IV elaborates on the proposed elliptical copula regression. Experimental results are provided in Section V. Section VI draws the conclusion of the work.

II. BACKGROUND ON MT SOURCE CODING

We consider a WSN comprising L sensors that collect data produced by correlated sources $X_1, X_2, \ldots, X_L$, which take values from L continuous alphabets $X_1, X_2, \ldots, X_L$ and are drawn i.i.d. according to the joint probability density function (pdf) $f_X(x_1, x_2, \ldots, x_L)$. Each sensor, indexed by $l \in \mathcal{I}_L = \{1, 2, \ldots, L\}$, gathers a sequence of n source samples and forms a data block $x_l = [x_l(1), x_l(2), \ldots, x_l(n)]$. The data are encoded using L separate encoding functions $\phi_l : X^n_l \rightarrow \{1, \ldots, 2^{nR_l}\}$, $l \in \mathcal{I}_L$, (1) where each compresses the source block $x_l$ at rate $R_l$ by assigning to it a discrete index $\phi_l(x_l)$. The joint decoder is a function

$\theta : \{1, \ldots, 2^{nR_1}\} \times \cdots \times \{1, \ldots, 2^{nR_L}\} \rightarrow \hat{X}_1^n \times \cdots \times \hat{X}_L^n$, (2)

that reconstructs the data blocks of all sensors, $[\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_L]$, based on the observed index tuple $[\phi_1(x_1), \phi_2(x_2), \ldots, \phi_L(x_L)]$.

Let $d_l(\cdot)$ be a distortion measure for the sensor $l$, defined as $d_l : \hat{X}_l \times X_l \rightarrow \mathbb{R}_+$. Given a distortion tuple $D = [D_1, D_2, \ldots, D_L]$, the rate tuple $R = [R_1, R_2, \ldots, R_L]$ is achievable if, for any $\epsilon > 0$, there exists a large enough $n$, L source encoder functions $\phi_l$, and a decoder function $\theta$ such that the distortion constraint $\frac{1}{n} \sum_{i=1}^{n} E_d[d(x_l(i), \hat{x}_l(i))] \leq D_l + \epsilon$ be satisfied for each $l \in \mathcal{I}_L$. The achievable rate-distortion region $\mathcal{R}^n(D)$ is the convex hull of all achievable rate tuples $R$.

Two code designs for the two-terminal Gaussian MT problem are proposed in [17], where TCQ is combined with SW coding. In the first scheme, labeled as asymmetric SW Coded Quantization (SWCQ), each data block of one source, say
\(X_1\), is quantized and entropy encoded so as to act as side information to encode the corresponding data block of \(X_2\) by means of WZ coding. Then the decoded information is linearly combined to produce side information that is used to further refine \(X_1\). Finally, the reconstructed data blocks, denoted by \(\hat{x}_1\) and \(\hat{x}_2\), respectively, are passed to a linear estimator that yields the final decoded estimates \([\hat{x}_1, \hat{x}_2]\). In the second scheme [17], referred to as symmetric SWCQ, data blocks produced by both sources are quantized and compressed using symmetric SW coding, based on the concept of channel code partitioning. At the decoder, symmetric SW decoding is followed by inverse quantization to reconstruct the two blocks. Similarly to asymmetric SWCQ, a linear estimation step is finally applied. However, extending the designs in [17] to multiple sources is challenging, as practical SW coding based on channel codes becomes difficult to implement.

To increase flexibility at the expense of compression performance, the authors of [19] studied the specific MT source coding scenario where SW coding is replaced with simple entropy coding. A practical realization of this scheme is presented in [20], where \(L\) sensors monitor homogeneous data types. Each encoder performs USQ followed by arithmetic encoding. At the decoder, after decoding the blocks \(\hat{x}_1\) from all sensors, a second estimation stage is applied, where the dependencies among the sensed data are exploited through Gaussian regression, as explained below.

**Gaussian Regression Stage.** Let the random vector \(X = [X_1, \ldots, X_L]\), which describes the data produced from all sensors at instant \(i \in \{1, 2, \ldots, n\}\), follow a multivariate normal distribution, \(X \sim \mathcal{N}(\mu_X, \Sigma_X)\), with mean value \(\mu_X\) and covariance matrix \(\Sigma_X\). Moreover, let the quantization noise \(Z_l\), which corrupts each component \(X_l\) in \(X\), be additive, independent of \(X_l\), and temporally independent. The dequantized data random variable at instant \(i\) from the \(l\)-th sensor is given by \(\hat{X}_l = X_l + Z_l\). The variance of the quantization noise \(Z_l\) can be calculated as

\[
\sigma^2_{Z_l} = \frac{\int_{x_l \in Q[x_l(i)]} (\hat{x}_l(i) - x_l)^2 f_{X_l}(x_l) dx_l}{\int_{x_l \in Q[x_l(i)]} f_{X_l}(x_l) dx_l},
\]

where \(Q[x_l(i)]\) and \(\hat{x}_l(i)\) are, respectively, the quantization index and the reconstructed value assigned to \(x_l(i)\), and \(f_{X_l}(x_l)\) is the marginal pdf of \(X_l\). The vectors \(X\) and \(\hat{X}\) are assumed to be jointly Gaussian, i.e.,

\[
\begin{bmatrix}
\hat{X} \\
X
\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}
\mu_{\hat{X}} \\
\mu_X
\end{bmatrix}, \begin{bmatrix}
\Sigma_{\hat{X}\hat{X}} & \Sigma_{\hat{X}X} \\
\Sigma_{X\hat{X}} & \Sigma_{XX}
\end{bmatrix}\right),
\]

where \(\Sigma_{XX} = \Sigma_{\hat{X}\hat{X}} = \Sigma_{\hat{X}\hat{X}}^T = \Sigma_X\), and \(\Sigma_{X\hat{X}} = \Sigma_Z\), with \(\Sigma_Z\) being a diagonal matrix with nonzero elements \(\sigma^2_{Z_l}\) and \((\cdot)^T\) denoting matrix transpose. Given the dequantized data \(\hat{X}\), the final estimate \(\hat{X}\) is given by the conditional mean \(\mu_{\hat{X}|\hat{X}}\) of \(X|\hat{X}\) as \(X|\hat{X} \sim \mathcal{N}(\mu_{X|\hat{X}}, \Sigma_{X|\hat{X}})\), that is, \([35]\)

\[
\hat{X} = \mu_{X|\hat{X}} + \Sigma_{X|\hat{X}}^{-1}(\hat{X} - \mu_{\hat{X}}).
\]

**III. THE PROPOSED MT SOURCE CODE DESIGN**

The architecture of the proposed MT source coding system is depicted in Fig. 1. Unlike prior studies, we assume that the sensors collect heterogeneous correlated data (e.g., temperature, humidity, and light). Furthermore, contrary to prior studies [19], [20], we assume that the consecutive data samples, collected by each sensor, are highly correlated. With the aim to make use of this temporal correlation, each sensor \(l \in I_L\) gathers a block of \(n\) readings, \(x_l = [x_l(1), x_l(2), \ldots, x_l(n)]\), and applies DPCM encoding [36], the block diagram of which is shown in Fig. 2. The DPCM encoder comprises a linear prediction function and a Lloyd-Max [6] scalar quantization function \(Q(\cdot)\). For the \(l\)-th sensor, the prediction value for the data sample at instant \(i\) is given by

\[
v_l(i) = \sum_{j=1}^{m} a_j x_l(i - j),
\]

where \(m\) is the memory length of the predictor. The coefficients \(a_j, j = 1, \ldots, m\), are chosen so as to minimize the MSE between \(x_l(i)\) and \(v_l(i)\), and are estimated by solving the Yule-Walker equations [36]. A Lloyd-Max scalar quantizer with \(M\) reconstruction levels is used to quantize the prediction error \(w_l(i) = x_l(i) - v_l(i)\). The reconstruction points and the quantization regions of the quantizer are determined during a training period. The \(l\)-th DPCM encoder outputs a block \(q_l = [Q(w_l(1)), Q(w_l(2)), \ldots, Q(w_l(n))]\) containing the quantization indices of the prediction errors \(w_l = [w_l(1), w_l(2), \ldots, w_l(n)]\). The DPCM block \(q_l\) is then arithmetic entropy encoded [37] at rate \(R_l = \frac{1}{n} H(q_l)\) bits/sample. Arithmetic coding is very efficient, allowing for a compression rate that is very close to the empirical entropy.

At the joint decoder, the bitstream received from each sensor \(l \in I_L\) is arithmetic entropy decoded, producing the quantization indices \(q_l\). The reconstructed values \(\hat{w}_l = [\hat{w}_l(1), \hat{w}_l(2), \ldots, \hat{w}_l(n)]\) of the prediction errors, are then calculated via inverse quantization. Subsequently, the decoder applies DPCM decoding per sensor for estimating the source blocks \(\hat{x}_l = [\hat{x}_l(1), \hat{x}_l(2), \ldots, \hat{x}_l(n)]\) for all sensors.

Upon reconstructing the source blocks, the joint decoder performs an additional estimation stage, where the dependence structure among the heterogeneous data collected by the various sensors is exploited. To express the joint statistics among diverse correlated data, the proposed system adheres to a modeling approach based on copula functions [31]. Namely,
the vector \( \hat{x}(i) = [\tilde{x}_1(i), \tilde{x}_2(i), \ldots, \tilde{x}_L(i)] \), containing the dequantized values from all sensors at instant \( i \), is passed to the proposed copula regression algorithm that outputs a refined version, denoted by \( \hat{x}(i) = [\tilde{x}_1(i), \tilde{x}_2(i), \ldots, \tilde{x}_L(i)] \). Fig. 3 presents the proposed DPCM decoder of each sensor \( l \).

The copula regression algorithm that we devise in this paper is described in the next section. The final estimates \( \hat{x}(i) \) are calculated for all \( i = 1, 2, \ldots, n \), and are used to estimate the source blocks \( \hat{x}_i = [\tilde{x}_1(1), \tilde{x}_2(2), \ldots, \tilde{x}_L(n)] \) for all sensors.

IV. THE PROPOSED SEMI-PARAMETRIC COPULA REGRESSION

Focusing on homogeneous data types, previous studies [13], [16], [20] express the inter-sensor data dependencies using the multivariate normal distribution. This implies that the marginal distributions are assumed normal and that the dependency among the data is considered to be linear. However, when dealing with heterogeneous information sources these assumptions may be inaccurate, due to, for example, variations in signal dimensionality across diverse modalities.

In support of this argument, Figs. 4(a) and 4(b) show that the marginal pdfs of the data collected by a temperature and a humidity sensor, respectively, do not follow a Gaussian distribution. In order to accurately express the dependencies among diverse data sources, we propose the use of statistical models based on copula functions [30], [31]. Copula functions can combine heterogeneous sensor data with disparate marginal distributions into a multivariate pdf.

A. Copula Functions

In the literature [31], [38]–[41], there exist various bivariate and multivariate copula families. In this work, we concentrate mainly on the most well-established of them, namely, the Elliptical and the Archimedean copula families. Elliptical copulas provide symmetric expressions and are suitable for applications where the number of the variables (i.e., \( L \)) increases. Typical examples of Elliptical copulas are the normal copula and the t-copula (alias, Student’s copula). It is worth mentioning that Elliptical copulas facilitate the design of regression schemes since they allow for a non-constant degree of association (i.e., the Spearman rank coefficient) between the response variable and the covariates [33]. Contrary to Elliptical copulas, Archimedean copulas are easily derived and capture a wide range of dependence. However, being parameterized by a single parameter, they lack some modeling flexibility. Among others, the Clayton copula is a widely used Archimedean copula, since it has a simple closed-form expression.

In our system, copula functions are used to model the statistical distribution of the random vector \( X = [X_1, \ldots, X_L] \) that describes the sensor values at instant \( i \) (see Fig. 2).

Definition. Let \( F_{X_1}(x_1), F_{X_2}(x_2), \ldots, F_{X_L}(x_L) \) be the continuous marginal cumulative distribution functions (cdfs) of the random variables in \( X \). By applying the probability integral transform [42] on each \( X_l, l \in I_L \), the vector \( X \) is transformed into the vector \( U = [U_1, U_2, \ldots, U_L] \), where \( U_l = F_{X_l}(X_l) \). Therefore, regardless of the marginal distribution of \( X_l \), the transformed variable \( U_l \) always follows a uniform distribution. According to Sklar’s theorem [30], [31], if \( F_X \) is the \( L \)-dimensional joint distribution of \( X \), there exists a unique \( L \)-dimensional copula function \( C : [0, 1]^L \to [0, 1] \) such that

\[
F_X(x_1, x_2, \ldots, x_L) = C(F_{X_1}(x_1), F_{X_2}(x_2), \ldots, F_{X_L}(x_L)).
\]

If the marginal distributions are continuous, an assumption that holds in our case, the copula function is unique.

Copula density function. The copula density function, denoted by \( c(F_{X_1}(x_1), \ldots, F_{X_L}(x_L)) \), can be found by differentiating the expression in (7) with respect to \( u_l = F_{X_l}(x_l) \), for all \( l \in I_L \). Thus, the multivariate pdf of the sensor data can be written as:

\[
f_X(x_1, \ldots, x_L) = c(F_{X_1}(x_1), \ldots, F_{X_L}(x_L)) \prod_{l=1}^{L} f_{X_l}(x_l).
\]

Given the marginal pdfs of the random variables, an appropriate copula function that best captures the dependencies among the sensor data should be selected. In this work, we consider the multivariate normal, t- and Clayton copulas.

Normal copula. The multivariate normal copula function [31] is defined as:

\[
C_{\sigma}(u) = \Phi_{R_{\sigma}}(\Phi^{-1}(u_1), \Phi^{-1}(u_2), \ldots, \Phi^{-1}(u_L)),
\]

where \( \Phi \) is the cumulative distribution function of the standard normal distribution, and \( R_{\sigma} \) is the correlation matrix.

Archimedean copula. The Archimedean copula function [31] is defined as:

\[
C(u) = \int_0^u c(t) dt
\]

where \( c(t) \) is a non-decreasing function on \([0, 1]\) that is the derivative of the copula function.

Clayton copula. The Clayton copula function [31] is defined as:

\[
C_{\theta}(u_1, u_2, \ldots, u_L) = (\sum_{i=1}^{L} u_i^{-\theta} - 1)^{-\frac{1}{\theta}}
\]

where \( \theta \) is a positive parameter that controls the degree of dependence.

ELLIPTICAL COPULA. The multivariate normal copula function [31] is defined as:

\[
C_{\sigma}(u) = \Phi_{R_{\sigma}}(\Phi^{-1}(u_1), \Phi^{-1}(u_2), \ldots, \Phi^{-1}(u_L)),
\]
where $\mathbf{u} = [u_1, \ldots, u_L]$ is a realization of the random vector $\mathbf{U}$, $\Phi_{R_\delta}$ denotes the standard multivariate normal distribution with linear correlation matrix $R_\delta$, and $\Phi^{-1}$ is the inverse function of the standard univariate normal distribution. The normal copula density is given by [43]

$$c_\delta(\xi) = |R_\delta|^{-\frac{1}{2}} \exp \left( -\frac{1}{2} \xi (R_\delta^{-1} - I) \xi^T \right),$$

where $\xi = [\Phi^{-1}(u_1), \Phi^{-1}(u_2), \ldots, \Phi^{-1}(u_L)]$ and $I$ is the $L \times L$ identity matrix.

**Student's t-copula.** The multivariate t-copula function $[31]$ has the form:

$$C_t(\mathbf{u}) = T_{R_t, \nu}(t_{\nu}^{-1}(u_1), t_{\nu}^{-1}(u_2), \ldots, t_{\nu}^{-1}(u_L)), \quad (11)$$

where $T_{R_t, \nu}$ is the standardized multivariate t-distribution with $\nu$ degrees of freedom and correlation matrix $R_t$, and $t_{\nu}^{-1}$ denotes the inverse of the univariate t-distribution. The t-copula density is [31]

$$c_t(\eta) = |R_t|^{-\frac{1}{2}} \Gamma \left( \frac{\nu+1}{2} \right) \left( 1 + \frac{1}{\nu} \eta R_t^{-1} \eta^T \right)^{-\frac{\nu+L}{2}} \Gamma \left( \frac{\nu}{2} \right) \prod_{l=1}^L \left( 1 + \frac{\eta_l^2}{\nu} \right)^{-\frac{\nu}{2}},$$

where $\eta = [\eta_1, \eta_2, \ldots, \eta_L]$, $\eta_l = t_{\nu}^{-1}(u_l), l \in \mathcal{I}_L$ and $\Gamma(\cdot)$ is the gamma function.

**Clayton copula.** An Archimedean copula $C^{(a)}$ can be represented by $[31]$:

$$C^{(a)}(\mathbf{u}; \delta) = \gamma^{-1}(\gamma(u_1; \delta) + \cdots + \gamma(u_L; \delta), \delta),$$

where $\gamma: [0, 1] \times \Delta \rightarrow [0, \infty)$ is a continuous, strictly decreasing and convex function such that $\gamma(1; \delta) = 0$, and $\delta$ is the copula parameter defined in the range $\Delta$. The function $\gamma(u_1, l \in \{1, \ldots, L\}$, is called generator and its pseudo-inverse, defined by:

$$\gamma^{-1}(u_l; \delta) = \begin{cases} \gamma^{-1}(u_l; \delta) & \text{if } 0 \leq u_l \leq \gamma(0; \delta), \\ 0 & \text{if } \gamma(0; \delta) \leq u_l \leq \infty, \end{cases}$$

has to be completely monotonic of order $L$ [44]. For the Clayton copula, the generator has the following form [31]

$$\gamma_{Cl}(u_1) = \delta_{Cl}^{-1}(u_1^{-\delta} - 1),$$

where $\delta_{Cl}$ is the parameter of the Clayton copula. The expression of the Clayton copula function is given by

$$C^{(Cl)}(\mathbf{u}) = \left( \sum_{l=1}^L u_l^{-\delta_{Cl}} - L + 1 \right)^{-1/\delta_{Cl}},$$

whereas the Clayton copula density function can be written as [45]:

$$c^{(Cl)}(\mathbf{u}) = \left( \sum_{l=1}^L u_l^{-\delta_{Cl}} - L + 1 \right)^{-L-1/\delta_{Cl}} \times \prod_{l=1}^L (u_l^{-\delta_{Cl} - 1}((l - 1)\delta_{Cl} + 1)).$$

The multivariate pdf of the sensor values is given by replacing the copula density $c(F_{X_l}(x_1), \ldots, F_{X_L}(x_L))$ in (8) with the expression in either (10), (12), or (17).

### B. The Proposed Copula Regression Method

The proposed code design embodies a copula regression method that takes as input the reconstructed sensor values $\hat{x}(i) = [\hat{x}_1(i), \hat{x}_2(i), \ldots, \hat{x}_L(i)]$ and produces a refined estimate, denoted by $\check{x}(i) = [\check{x}_1(i), \check{x}_2(i), \ldots, \check{x}_L(i)]$.

During a training stage, the model parameters, namely, either the correlation matrix $R_\delta$ of the normal copula, the correlation matrix $R_t$ and the degrees of freedom $\nu$ of the t-copula, or the parameter $\delta_{Cl}$ of the Clayton copula, as well as the continuous marginal pdfs and cdfs of sensor data are estimated. The correlation matrix $R_\delta$ of the normal copula is parametrically estimated using standard Maximum Likelihood Estimation (MLE) [46]. Regarding the t-copula, the correlation matrix $R_t$ and the degrees of freedom parameter $\nu$ are parametrically estimated using approximate MLE [46], where the copula parameter is fitted by maximizing an objective function that approximates the profile log-likelihood for the degrees of freedom parameter. Moreover, the parameter of the Clayton copula is estimated using MLE.

The marginal pdfs are non-parametrically estimated using kernel density estimation (KDE) [47]. Specifically, given $\tau$ training samples from sensor $l \in \mathcal{I}_L$, the KDE estimator is

$$\hat{f}_{X_l}(x_l) = \frac{1}{\tau h_{X_l}} \sum_{h=1}^\tau K \left( \frac{x_l - x_{h_l}}{h_{X_l}} \right),$$

where $K(\cdot)$ is the kernel function and $h_{X_l}$ is the bandwidth of the smoothing window. The kernel function is usually chosen to be a smooth unimodal function with a peak at zero. In the literature [47] various kernel functions, such as the Gaussian and the Epanechnikov kernel, have been proposed. Although the Epanechnikov kernel is optimal in the MSE sense [48], the accuracy of the non-parametric estimate depends less on the shape of the kernel function $K(\cdot)$ than on the value of its bandwidth $h_{X_l}$ [49]. For accurate density estimation, an appropriate selection of the bandwidth value is important since small or large values can lead to under- or over-smoothed estimators, respectively. In our implementation, the optimal bandwidth value is obtained by the MATLAB function ksdensity, which uses a rule of thumb [47]. The smooth estimate of the corresponding marginal cdf, $\hat{F}_{X_l}$, is constructed by integrating $\hat{f}_{X_l}$. That is,

$$\hat{F}_{X_l}(x_l) = \int_{x_l}^{\infty} \hat{f}_{X_l}(x) dx = \frac{1}{\tau} \sum_{h=1}^\tau \kappa \left( \frac{x_l - x_{h_l}}{h_{X_l}} \right),$$

where $\kappa(x) = \int_{x}^{\infty} K(\chi) d\chi$.

This work focuses on slowly-varying data sources, such as temperature and humidity, which are modelled as stationary processes, namely, random processes whose statistical properties are time independent or do not change for a given period of time [36]. Thus, the entries of the correlation matrices $R_\delta$ (for the normal copula) or $R_t$ (for the t-copula), as well as the parameter of the Clayton copula, are assumed to be

2Note that vector $\xi$ is a function of $\mathbf{u}$. Thus it can be written either $c_\gamma(\xi)$ or $c_\gamma(\mathbf{u})$. 

[43] [44] [45] [46] [47] [48] [49]
constant for a given period of time and offline estimation is sufficient. A similar approach has been followed in previous works [15], [20]. Nevertheless, the use of online estimators for the statistical parameters is left for future research.

Once the parameters of the models have been determined, the refined estimates $\hat{x}(i)$ are calculated based on the proposed regression algorithm. The algorithm refines the reconstructed sensor value $\hat{x}(i)$, corresponding to the $l$-th sensor, using the other elements of $\hat{x}(i)$, which correspond to the other sensors. Let the response variable be $X_l$ and the variables $\{X_c : c \in I_L \setminus \{l\}\}$ denote the covariates. Existing copula regression methods [32], [33], would estimate the values in $x(i)$ using the conditional mean $E[X_l|X_1, \ldots, X_i], c \in I_L \setminus \{l\}$. This approach works well when the number of covariates is small (two or three). However, when the dimensionality is high, as in our system, exact inference can be intractable, thereby requiring computationally expensive Monte Carlo sampling methods [50].

Another approach for predicting the refined estimates in the vector $\hat{x}(i)$ considers the MLE problem, which can be written as

$$\hat{x}_l = \arg \max_{x_l} f_{X_l|X_1}(x_1, \ldots, x_L|x_l)$$

$$= \arg \max_{x_l} \left[ c(F_{X_l}(x_1), \ldots, F_{X_L}(x_L)) \times \frac{f_{X_l}(x_l)}{f_{X_1}(x_1) \prod_{k=1}^L f_{X_k}(x_k)} \right]$$

$$= \arg \max_{x_l} c(F_{X_l}(x_1), \ldots, F_{X_L}(x_L)) \prod_{k=1}^L f_{X_k}(x_k)$$

$$= \arg \max_{x_l} c(F_{X_l}(x_1), \ldots, F_{X_L}(x_L)),$$  \hspace{1cm} (20)

where $X^c = \{X_1, X_2, \ldots, X_c\}, c \in I_L \setminus \{l\}$. Eq. (20) is derived based on Bayes’ rule $f_{X_l|X_1}(x_1, \ldots, x_L|x_l) = f_{X_l}(x_l|x_1, \ldots, x_L)$ and the expression in (8) that provides the description of the multivariate pdf $f_{X}(x_1, \ldots, x_L)$. In this work, we solve (21) by using an algorithm that delivers accurate inference at reasonable complexity. In particular, the cdf of the response variable $F_{X_l}(x_l)$ is sampled until the considered copula density $c_\rho(\xi), c_\eta(\eta)$ or $c_{\text{Clay}(\mathbf{u})}$ is maximized. This is expressed by the following optimization problem:

$$\hat{u}_l = \arg \max_{u_l = F_{X_l}(x_l)} c(u_1, \ldots, u_L, u_L),$$  \hspace{1cm} (22)

where $u_l \in [0, 1]$ and $c(\cdot)$ is replaced by the expression in either (10), (12) or (17). The solution of (22) is found numerically using a greedy approach. The objective function in (22), is not necessarily concave meaning that local maxima may appear. We solve problem (22) using a greedy approach to find the optimal $u_l$, which does not abide by the “hill-climbing principles” [51] that lead to calculation of local maxima; the algorithm rather performs an exhaustive search for all values of $u_l$ that span the region $[0, 1]$ and finds the global maximum (within the step-size accuracy) of the copula density. The sampling step is chosen to be $u_0 = 0.001$ such that we strike a balance between: (a) the decoding complexity level, where larger parameter values speedup the optimization process, and (b) the accuracy of the inference, where smaller values provide

meticulous copula sampling. Finally, the refined estimate of the sensor value is given by

$$\hat{x}_l(i) = \tilde{F}_{X_l}^{-1}(u_l^*_i).$$  \hspace{1cm} (23)

The procedure is described in Algorithm 1. Initially, the algorithm refines the value of the $l$-th sensor using the other elements of $\hat{x}(i)$, which correspond to dequantized values of the remaining sensors. Subsequently, the algorithm replaces the corresponding dequantized estimate $\hat{x}_l(i)$ with the refined value $\hat{x}_l(i)$ in $\hat{x}(i)$ and continues with refining the value of the next sensor. The same procedure repeats for all unrefined symbols in $\hat{x}(i)$, yielding the refined estimate vector $\hat{x}(i)$. The sensors indices are processed sequentially with increasing the value of $l$. Moreover, as shown in Algorithm 1, the proposed methodology can be straightforwardly adapted to cope with the case where a subset of sensors, denoted by $I_c$, is not operating (due to, for example, battery depletion or duty cycling for extending the lifetime of the system)\(^3\). In this case, the vector $\hat{x}_c(i)$ contains only the components of $\hat{x}(i)$ that correspond to the effective sensors, which are indexed in the set $I_c$. Furthermore, only the columns and rows of the correlation matrices $R_l$ (for the normal copula) or $R_h$ (for the $t$-copula) that correspond to the effective sensors are kept and the remaining rows and columns are removed.

V. EXPERIMENTAL EVALUATION

We evaluate the proposed system using actual sensor readings from the Intel-Berkeley Research lab [34] database. The database contains data collected from 54 Mica2Dot sensors equipped with weather boards, monitoring diverse physical parameters (that is, humidity, temperature, and light) in an indoor office and laboratory environment. To conduct our experiments, we selected randomly $L = 21$ sensors from the database, 11 of which harvest temperature data (in °C) and the other ten collect humidity data (in %). The sensor readings from the Intel-Berkeley database exhibit various levels of dependence, such as strong ($\rho_{1,3} = 0.9764$), medium ($\rho_{8,18} = 0.7574$) and weak ($\rho_{9,14} = 0.5311$), where $\rho_{i,j}$ denotes Spearman’s rank correlation coefficient between data of sensors $l_1$ and $l_2$.

The collected data were distinguished into a training and an evaluation set, without an overlap between the two. The former, which consisted of the initial 15% of the data, was used to derive the parameters of the proposed coding scheme and the proposed copula-function-based model. Given the training dataset, we derived two different predictors of memory length $m = 3$: one for the sensors measuring temperature and one for those sensing humidity. The derived predictor coefficients, which led to minimum-MSE predictors [see (6)], were found to be $a_{1,1} = 2.7302, a_{1,2} = -2.6647, a_{3,3} = -0.9127$, for the temperature data, and $a_{h,1} = 1.1613, a_{h,2} = -0.0490, a_{h,3} = -0.1124$ for the humidity data. Furthermore, following the semi-parametric approach described in Section IV, we estimated the correlation matrix $R_p$ of the normal copula density, the correlation matrix $R_h$ and the degrees of freedom $\nu$ for the $t$-copula density, the parameter of the Clayton copula

\(^3\)A subset of sensors is periodically turned off during specific time periods.
Algorithm 1 Proposed copula regression algorithm for refining the dequantized sensor values.

1. **Inputs**: Set of effective sensors $I_e = \{i_1, \ldots, i_n\}$, vector $\tilde{x}_e(i)$ with dequantized sensor values in $I_e$, copula parameters, marginal statistics.
2. **Output**: Refined estimates $\tilde{x}_e(i)$.
3. Modify $R_q$ (or $R_c$) given the set $I_e = \{i_1, \ldots, i_n\}$. 
4. for $l \in I_e$ do 
5. $c_l^* = 0$; 
6. $u_l^* = 0$; 
7. for $u_l = 0 \rightarrow 1$ with step $u_{sl}$ do 
8. Create the vector $\mathbf{u} = [\tilde{F}_{X_1}(\tilde{x}_1(i)), \ldots, u_l, \ldots, \tilde{F}_{X_n}(\tilde{x}_n(i))]$. 
9. if Elliptical copula then 
10. Calculate $\xi = \phi^{-1}(\mathbf{u})$ (or $\eta = \tau^{-1}(\mathbf{u})$).
11. Calculate $c_l(\xi)$ using (10) or $c_l(\eta)$ using (12).
12. if $c_l(\xi) \geq c_l^*(\mathbf{u})$ (or $c_l(\eta) \geq c_l^*(\mathbf{u})$) then 
13. $c_l^* = c_l(\xi)$ (or $c_l^* = c_l(\eta)$); 
14. $u_l^* = u_l$;
end if 
else 
17. Calculate $c_l^{(c)}(\mathbf{u})$ using (17).
18. if $c_l^{(c)}(\mathbf{u}) \geq c_l^*$ then 
19. $c_l^* = c_l^{(c)}(\mathbf{u})$; 
20. $u_l^* = u_l$;
end if 
end if 
23. end for 
24. Calculate $\tilde{x}_l(i) = \tilde{F}_{X_l}^{-1}(u_l^*)$. 
25. Replace $\tilde{x}_l(i)$ with $\tilde{x}_l(i)$ in $\tilde{x}_e(i)$.
26. end for 
27. Do $\tilde{x}_e(i) = \tilde{x}_e(i)$. 

$
\delta_{cl}$, as well as the marginal pdfs $\tilde{f}_{X_l}$ and cdfs $\tilde{F}_{X_l}$ on the sensor values. To compare the proposed modeling approach against the state of the art [20], we also fitted the multivariate Gaussian model on the data; namely, we estimated parametrically$^4$ the mean values $\mu_{X_l}$ and the standard deviations $\sigma_{X_l}$ of the marginal distributions for the sensor values as well as the corresponding covariance matrix $\Sigma_{X_l}$. The estimated parameters for the different statistical models are reported in Table I. The degrees-of-freedom parameter of the t-copula function model was found to be $\nu = 6.6995$. Finally, we calculated the parameter of the Clayton copula via MLE, which was found to be $\delta_{cl} = 0.5302$.

During the training stage, we also configure the Lloyd-Max quantizer and the arithmetic entropy coder that are deployed to compress the data from each sensor in our system. Specifically, we determine the reconstruction values and the partitions of each quantizer, as well as the source statistics used in the arithmetic coders.

During the evaluation stage, the data from each sensor $l \in I_{\infty}$ were aggregated into data blocks, each consisting of $n = 40$ consecutive samples. The block length is chosen to strike a balance between good compression performance and delay. The quantizer of each sensor uses the same number of quantization levels, $M$, resulting in a stream of $k = n \times \log_2 M$ bits that is passed to each entropy encoder.

### Table I

<table>
<thead>
<tr>
<th>Sensor ID</th>
<th>Mean $\mu_{X_l}$</th>
<th>St. dev $\sigma_{X_l}$</th>
<th>Bandwidth $\delta_{X_l}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (Hum.)</td>
<td>23.2250</td>
<td>4.1638</td>
<td>0.9315</td>
</tr>
<tr>
<td>2 (Hum.)</td>
<td>35.9024</td>
<td>7.8912</td>
<td>1.6196</td>
</tr>
<tr>
<td>3 (Temp.)</td>
<td>22.0690</td>
<td>2.1606</td>
<td>0.7681</td>
</tr>
<tr>
<td>4 (Hum.)</td>
<td>39.4034</td>
<td>5.1733</td>
<td>0.9993</td>
</tr>
<tr>
<td>5 (Temp.)</td>
<td>21.8954</td>
<td>5.2414</td>
<td>0.6403</td>
</tr>
<tr>
<td>6 (Hum.)</td>
<td>58.5627</td>
<td>5.3227</td>
<td>0.9814</td>
</tr>
<tr>
<td>7 (Temp.)</td>
<td>22.2108</td>
<td>4.2620</td>
<td>0.6094</td>
</tr>
<tr>
<td>8 (Hum.)</td>
<td>35.3782</td>
<td>5.7565</td>
<td>0.4494</td>
</tr>
<tr>
<td>9 (Temp.)</td>
<td>22.0934</td>
<td>2.8939</td>
<td>1.1969</td>
</tr>
<tr>
<td>10 (Hum.)</td>
<td>30.9978</td>
<td>5.8755</td>
<td>0.5175</td>
</tr>
<tr>
<td>11 (Temp.)</td>
<td>22.2791</td>
<td>5.3055</td>
<td>1.0858</td>
</tr>
<tr>
<td>12 (Hum.)</td>
<td>46.7514</td>
<td>6.4600</td>
<td>1.0884</td>
</tr>
<tr>
<td>13 (Temp.)</td>
<td>22.0608</td>
<td>2.9606</td>
<td>0.5910</td>
</tr>
<tr>
<td>14 (Hum.)</td>
<td>37.3404</td>
<td>5.4444</td>
<td>1.0532</td>
</tr>
<tr>
<td>15 (Temp.)</td>
<td>21.4999</td>
<td>2.7720</td>
<td>0.4276</td>
</tr>
<tr>
<td>16 (Hum.)</td>
<td>57.5436</td>
<td>5.9022</td>
<td>0.7690</td>
</tr>
<tr>
<td>17 (Temp.)</td>
<td>21.1885</td>
<td>2.9714</td>
<td>0.4655</td>
</tr>
<tr>
<td>18 (Hum.)</td>
<td>38.4008</td>
<td>1.8404</td>
<td>0.7994</td>
</tr>
<tr>
<td>19 (Temp.)</td>
<td>20.5123</td>
<td>2.9205</td>
<td>0.5148</td>
</tr>
<tr>
<td>20 (Hum.)</td>
<td>40.0336</td>
<td>5.5524</td>
<td>0.9118</td>
</tr>
<tr>
<td>21 (Temp.)</td>
<td>20.8536</td>
<td>2.5813</td>
<td>0.3595</td>
</tr>
</tbody>
</table>

Fig. 5. Fitting of non-parametric functions on temperature data collected by the Intel-Berkeley database [34], where the (a) Gaussian, (b) Laplacian, (c) Box, and (d) Epanechnikov kernels have been used.

### A. Choice of the Appropriate Kernel Function

First, we evaluate the impact of the fitting accuracy when different kernel functions are considered for the non-parametric estimation of the marginal pdfs. Fig. 5 depicts the fitting accuracy of different non-parametric distributions on the temperature data collected by sensor 1. The distributions use the Gaussian, the Laplacian, the Box and the Epanechnikov kernels. Moreover, using Kolmogorov-Smirnov fitting tests, Table II shows that the fitting accuracy of the different distributions is quite similar; this agrees with prior results such...
TABLE II
ASYMPTOTIC p-VALUES WHEN PERFORMING KOLMOGOROV-SMIRNOV FITTING TESTS BETWEEN THE ACTUAL TEMPERATURE READINGS AND THE SETS PRODUCED BY THE NON-PARAMETRIC DISTRIBUTION WITH DIFFERENT KERNELS. THE SIGNIFICANCE LEVEL IS SET TO 5%.

<table>
<thead>
<tr>
<th>Kernel Function</th>
<th>Bandwidth</th>
<th>p-values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian</td>
<td>0.4512</td>
<td>0.3792</td>
</tr>
<tr>
<td>Laplacian</td>
<td>0.4532</td>
<td>0.3351</td>
</tr>
<tr>
<td>Box</td>
<td>0.4532</td>
<td>0.2917</td>
</tr>
<tr>
<td>Epanechnikov</td>
<td>0.4532</td>
<td>0.3558</td>
</tr>
</tbody>
</table>

TABLE III
EFFECTIVE RATE VS. EFFECTIVE DISTORTION FOR GAUSSIAN, LAPLACIAN, BOX AND EPANECHNIKOV KERNEL FUNCTIONS (L_e = 21).

<table>
<thead>
<tr>
<th>Eff. Rate</th>
<th>Eff. Distortion</th>
</tr>
</thead>
<tbody>
<tr>
<td>(all schemes)</td>
<td>Gaussian</td>
</tr>
<tr>
<td>1.1161</td>
<td>1.3316</td>
</tr>
<tr>
<td>1.8876</td>
<td>0.3306</td>
</tr>
<tr>
<td>2.7818</td>
<td>0.1035</td>
</tr>
<tr>
<td>3.7408</td>
<td>0.0324</td>
</tr>
<tr>
<td>4.7228</td>
<td>0.0057</td>
</tr>
<tr>
<td>5.6577</td>
<td>0.0019</td>
</tr>
</tbody>
</table>

as [49], where it has been proven that the choice of the kernel function is less significant than the appropriate choice of the bandwidth of the smoothing window.

Table III shows the performance of the proposed system for different kernels. For illustrative purposes, we have assumed the normal copula regression for the refinement stage. The compression performance is expressed in terms of the effective distortion (in MSE), \( \frac{1}{n} \sum_{l \in L} D_{l} \), versus the effective rate (in bits/sample), \( \frac{1}{n} \sum_{l \in L} R_{l} \), for different quantization levels, that is, \( M = 8, 16, 32, 64, 128, \) or \( 256 \). Here we have assumed that \( L_e = 21 \), namely, all considered sensors are active. We see that, for all different kernel types, the effective distortion performance is very similar. Nevertheless, the Gaussian kernel provides slightly better results than the other functions and, hence, is used in our experiments.

B. Performance Evaluation of DPCM

We assess the impact of DPCM (including Max-Lloyd quantization) on the performance of the system. Particularly, we compare the system in [20], which applies USQ followed by arithmetic encoding, against our approach, which deploys DPCM with Lloyd-Max scalar quantization and arithmetic entropy encoding. In both systems, a Gaussian regression step is performed at the decoder after inverse quantization. The comparisons are conducted for two scenarios. In the first, all sensors are active, whereas in the second a random subset of sensors is may not operate, resulting in a setup with \( L_e \) effective sensors (see Section IV-B).

Figs. 6(a) and 6(b) illustrate the performance of the compared systems for \( L_e = 21 \) and \( L_e = 12 \), respectively. As in Section V-A, the compression performance is expressed in terms of the effective distortion versus the effective rate for different quantization levels. It is clear that the proposed approach leads to a substantially higher compression performance, delivering a significant effective rate reduction of up to 36.64% (when \( L_e = 21 \)) and 38.35% (when \( L_e = 12 \)) for a similar distortion level. These results underline the benefit of using a DPCM scheme with an optimized Lloyd-Max quantizer to leverage the intra-sensor dependencies in our setting.

C. Performance Evaluation of the Copula Regression Algorithm

We now evaluate the performance improvement achieved by the proposed copula-based regression algorithm. To this end, we remove the DPCM component from the system; namely, the collected data are quantized with the Lloyd-Max quantizer and the indices are entropy encoded. In particular, we compare the following schemes: (a) the baseline scheme using entropy coding without a refinement stage (i.e., no regression), (b) the scheme in [20] that combines entropy coding and Gaussian regression, (c) the proposed scheme using entropy coding and normal copula regression, (d) the proposed scheme that combines entropy coding and \( t \)-copula regression, and (e) the...
The proposed scheme that combines entropy coding and Clayton copula regression.

The effective rate-distortion performance of the system is given in Figs. 7(a) and 7(b) for $L_e = 21$ and $L_e = 12$, respectively. It is worth observing that the Gaussian regression method in [20] induces a higher distortion of the decoded data compared to the simple case where no regression is applied, especially at low rates. This is because in the low-rate regime the vectors $\mathbf{X}$ and $\hat{\mathbf{X}}$ in (4) are not jointly Gaussian and thus, Gaussian regression leads to poor final estimates. When the rate increases, the assumption of joint Gaussian distribution becomes more accurate and therefore better estimates are provided. However, the proposed copula regression algorithm systematically outperforms the Gaussian regression scheme in [20] for all copula models. More importantly, the improvements are increased when the encoding rate decreases. The reason is that at low rates, where the quantization of the data is coarse, the copula regression schemes offer significant improvement on the reconstruction quality.

Table IV presents the average percentage distortion reductions obtained by comparing each of the schemes (a), (c)-(e) with the state-of-the-art scheme (b). The improvements in distortion reduction refer to the cases when $L_e = 12$ and $L_e = 21$. These improvements show that copula-based models express the joint statistics among heterogeneous data more accurately than the multivariate Gaussian model. Furthermore, the $t$-copula function results in higher modeling accuracy than the normal copula, which is attributed to the ability of the former to express better the dependencies between extreme values [52]. Finally, the best performance is obtained when the Clayton copula is used because this copula efficiently captures the asymmetric dependencies among sensor data.

### Table IV

<table>
<thead>
<tr>
<th>Regression Type</th>
<th>$L_e = 12$</th>
<th>$L_e = 21$</th>
</tr>
</thead>
<tbody>
<tr>
<td>No regression [system (a)]</td>
<td>4.96%</td>
<td>39.07%</td>
</tr>
<tr>
<td>Normal copula regression [system (c)]</td>
<td>47.05%</td>
<td>57.31%</td>
</tr>
<tr>
<td>$t$-copula regression [system (d)]</td>
<td>65.93%</td>
<td>75.90%</td>
</tr>
<tr>
<td>Clayton copula regression [system (e)]</td>
<td>79.22%</td>
<td>87.55%</td>
</tr>
</tbody>
</table>

In the previous experiments, the proposed method was evaluated using actual sensor readings from the Intel Berkeley database, which are collected per minute; we refer to this data as dataset A. Due to the high sampling rate, consecutive sensor readings are highly correlated and, in this case, DPCM provides significant rate savings, as shown in Sections V-B and V-D. In order to assess the performance of the proposed method for weak intra-sensor dependencies, we perform a downsampling of the data in the Intel Berkeley database, namely, we consider sensor readings collected per 30 minutes; we refer to them as dataset B. Fig. 9 compares the autocorrelation function (ACF) of the temperature readings of sensor 1 for the datasets

### E. Performance Evaluation for Weaker Intra-Sensor Dependence Structure

In the previous experiments, the proposed method was evaluated using actual sensor readings from the Intel Berkeley database, which are collected per minute; we refer to this data as dataset A. Due to the high sampling rate, consecutive sensor readings are highly correlated and, in this case, DPCM provides significant rate savings, as shown in Sections V-B and V-D. In order to assess the performance of the proposed method for weak intra-sensor dependencies, we perform a downsampling of the data in the Intel Berkeley database, namely, we consider sensor readings collected per 30 minutes; we refer to them as dataset B. Fig. 9 compares the autocorrelation function (ACF) of the temperature readings of sensor 1 for the datasets...
Fig. 8. Rate-distortion performance comparison between the state-of-the-art system presented in [20] and the proposed system using DPCM and (normal, t- and Clayton) copula regression at the decoder. The number of effective sensors is (a) $L_e = 21$ and (b) $L_e = 12$.

Fig. 9. Autocorrelation function calculated for the temperature readings of sensor 1 during the training period, when data sampling is performed per (a) 1 minute, or (b) 30 minutes.

A and B. It is clear that the ACF decays faster for the dataset B, as the sampling interval is larger.

We compare the performance of the proposed system with DPCM and Clayton copula regression against the state-of-the-art system in [20] for dataset B. Moreover, in the comparison, we include the system that applies entropy encoding and Clayton copula regression. The reason for choosing the Clayton copula for regression is that it delivers the best MSE performance, as shown in Sections V-C and V-D. For the dataset B, the Clayton copula parameter was found to be $\delta_{Cl} = 0.5870$.

The effective rate-distortion performance of all schemes is given in Fig. 10, for $L_e = 21$. The results reveal that even when the intra-sensor dependence is weak, the proposed method outperforms the system in [20]. In particular, average rate savings of 1.1493 bits/sample are obtained, whereas the average distortion reductions are of 89.65%. The rate gain due to DPCM is smaller than in Section V-D, where the dataset A is considered, but still significant. Furthermore, the system with entropy coding and Clayton copula regression delivers better performance in MSE than the state-of-the-art system in [20], obtaining average reduction is 88.03%. Thus, the proposed copula regression algorithm delivers more accurate inference than the Gaussian regression [20] since copulas can efficiently capture the dependencies among the sensor data.

F. Comparison with DSC for Different WSN Topologies

Finally, we compare our system, which applies DPCM and Clayton copula regression, with a state-of-the-art DSC
The DSC system in [53], [54] abides by the following WSN topology [15]: the sensors are separated into smaller groups called clusters. Each cluster contains a cluster head (CH) and a number of peripheral nodes (PNs). The data collected from the CH are intra-encoded and communicated to a central node (decoder), where it plays the role of side information that is used to decode the data from the PNs, which are Wyner-Ziv [8] encoded. The readings of each sensor are uniformly quantized and the resulting quantization indices are split into bit-planes. The CH performs arithmetic entropy encoding of the bit-planes sequentially starting from the most significant one. The PNs perform Slepian-Wolf [7] encoding of the bit-planes using Low-Density Parity-Check Accumulate (LDPCA) codes [55]. A multivariate Gaussian distribution is used to describe the statistical dependencies among the sensor readings of the CH and the PNs.

We assume two different configurations of this topology. First, all 21 sensors form a big cluster with sensor 1 being the CH; we refer to this as a single-cluster topology. Second, the WSN is divided into two clusters; the first comprises 11 sensors measuring temperature, whereas the second includes the sensors that monitor humidity.

Fig. 11 shows that the proposed design outperforms significantly the DSC system for both the single-cluster and the two-cluster topologies. In particular, our system offers average rate savings of 1.2662 bits/sample and of 1.0088 bits/sample compared to the single- and two-cluster DSC system, respectively. The corresponding average effective distortion reductions are 90.14% and 90.25%. Thus, the proposed design efficiently leverages the inter-sensor dependencies among heterogeneous data, i.e., both temperature and humidity. However, we see that the two-cluster DSC system outperforms the single-cluster configuration, showing that this DSC design is more suitable for homogeneous data. Hence, grouping of sensors measuring homogeneous data (i.e., temperature or humidity) is required so as to improve its performance.

VI. CONCLUSION

We have proposed a novel MT source code design for multisensory WSNs monitoring diverse data, such as temperature and humidity. Our design achieves significant compression gains compared to the state of the art, because it takes into account both the inter- and intra-sensor data dependencies. Firstly, to express the dependence structure among the diverse data types collected by the various sensors (such as humidity or temperature sensors), we proposed the use of multivariate copula functions belonging to the Elliptical and the Archimedean family. Our system provides accurate statistical inference via regression, by means of a proposed algorithm that delivers accurate estimation at a reasonable complexity. Secondly, to leverage the intra-sensor data dependencies, we used a predictive quantization technique, namely, DPCM. Through experimentation using actual sensor measurements from the well-established Intel-Berkeley database [34], we showed that the proposed system significantly outperforms state-of-the-art designs [20], [53], [54]. Finally, the proposed scheme is flexible, as it does not require reconfiguration when a subset of sensors is not operating.

ACKNOWLEDGEMENTS

The work was supported in part by the FWO under Project G025615, in part by the VUB-UPatras International Joint Research Group on ICT (JICT) under Project OZR/2015/167, and in part by the VUB strategic research programme M3D2. We would also like to thank the reviewers for their constructive comments.

REFERENCES


M. P. Wand and M. C. Jones, "Dependence structures.

V. A. Epanechnikov, "Non-parametric estimation of a multivariate probability density.


R. T. Clemen and T. Reilly, "Correlations and copulas for decision and risk analysis.

H. Joe, "Copula modeling with copulas.

C. Genest and J. Mackay, "The joy of copulas: bivariate distributions with uniform marginals.


R. T. Clemen and T. Reilly, "Correlations and copulas for decision and risk analysis.


H. Joe, "Copulas for finance-a reading guide and some applications.

A. Munteanu currently serves as Associate Editor for IEEE Transactions on Multimedia.
Nikos Deligiannis (S08, M10) is assistant professor with the Electronics and Informatics department at Vrije Universiteit Brussel. He received the Diploma in electrical and computer engineering from University of Patras, Greece, in 2006, and the PhD in applied sciences (awarded with Highest Honors and congratulations from the jury) from Vrije Universiteit Brussel, Belgium, in 2012.

From Jun. 2012 to Sep. 2013, he was a postdoctoral researcher with the Department of Electronics and Informatics, Vrije Universiteit Brussel. From Oct. 2013 to Feb. 2015, he was working as postdoctoral research associate at the Department of Electronic and Electrical Engineering at University College London, UK. During that period, he was also acting as technical consultant on big visual data technologies at the British Academy of Film and Television Arts (BAFTA), UK. His current research interests include big data processing, compressed sensing, data analysis and networking for the internet of things, distributed computing, and visual search. Prof. Deligiannis has authored over 70 journal and conference publications, book chapters, and two patent applications (one owned by iMinds, Belgium and the other by BAFTA, UK). He was a recipient of the 2011 ACM/IEEE International Conference on Distributed Smart Cameras Best Paper Award and the 2013 Scientific Prize FWO-IBM Belgium.