Robust Hypersphere Fitting from Noisy Data Using Gibbs Sampling

Younes Boutiyarzist
University of Toulouse
TéSA Laboratory
Toulouse, France
younes.boutiyarzist@tesa.prd.fr

Julien Lesouple
University of Toulouse
ENAC
Toulouse, France
julien.lesouple@enac.fr

Jean-Yves Tourneret
University of Toulouse
INP-ENSEEIHT/IRIT/TéSA
Toulouse, France
jean-yves.tourneret@toulouse-inp.fr

***Abstract*—This paper studies a robust algorithm allowing the estimation of the center and the radius of a hypersphere in the presence of outliers. To that extend, the Student-t distribution is assigned to the noise samples to mitigate the impact of the outliers. A von Mises-Fisher prior distribution is also assigned to latent variables in order to exploit the fact that the observed samples are located in a part of the hypersphere. A robust Bayesian algorithm based on a Gibbs sampler is then proposed to solve the hypersphere fitting problem. This algorithm generates samples asymptotically distributed according to the joint distribution of the unknown parameters of the hypersphere (radius and center), as well as the other model parameters such as the noise variance. Simulations conducted on synthetic data with controlled ground truth allow the performance of this algorithm to be appreciated.**

***Index Terms*—t-distribution, von Mises-Fisher distribution, Gibbs sampler, hypersphere fitting, robust estimation**

I. Introduction

In the context of a biological study of phenotype evolution, LiDAR sensors capturing the same scene from several angles have been mounted on a phenoMobile [1]. The calibration of these sensors is made using spheres whose parameters have to be estimated for each LiDAR sensor [2]. This problem is related to the adjustment of hyperspheres from noisy point clouds referred to as hypersphere fitting, which is recurrent in many applications such as object tracking [3]–[5], robotics [6]–[8] or pattern recognition [9]–[11].

A specificity of the problem addressed in this paper is that only part of the hypersphere is reached by the LiDAR point cloud, which makes standard fitting methods inefficient. This problem was studied in [2] that derived an interesting EM algorithm for hypersphere fitting. A robust implementation of the EM algorithm was also investigated in [12] using a mixture model allowing inliers and outliers to be detected. The first component of this mixture model was based on latent variables defined as affine transformations of unitary random vectors having von Mises-Fisher distributions, while the second component used latent variables with Bernoulli distributions indicating whether each point is an outlier or not. The von Mises-Fisher distribution is a probability distribution on a hypersphere in $\mathbb{R}^d$. The parameters of this distribution are the mean direction $\boldsymbol{\mu}$, which is a unit vector defining the direction of the point cloud, and the concentration parameter κ, which represents how much the data move away from the mean direction. This distribution reduces to the uniform distribution over the hypersphere when $\kappa = 0$. It is well adapted to data that are concentrated on a part of a hypersphere when $\kappa > 0$ (e.g., when a LiDAR is pointing a sphere). The estimation method presented in [12] is very robust to outliers located uniformly in a fixed window containing the sphere of interest. However, the performance of the method drops for other kinds of outliers, which can be observed in some practical applications. This paper introduces a robust Student's t-distribution for the additive noise corrupting the hypersphere measurements, which overcomes this problem. The Student's t-distribution is classically used to ensure robust estimation due to its heavy-tailed nature. As closed-form expressions for the Bayesian estimators of this model are intractable, a Gibbs sampler is investigated generating samples that are used to estimate the model parameters.

This paper is organized as follows. Section 2 introduces the model used for robust hypersphere fitting and derives its posterior distribution and the conditional distributions required for the Gibbs sampler. Section 3 presents the different steps of the proposed Gibbs sampler. Simulations conducted on synthetic data are studied in Section 4 whereas conclusions are reported in Section 5.

II. Bayesian Model

A. Problem formulation

Consider n noisy measurements $\boldsymbol{z}_i \in \mathbb{R}^d$, $i = 1, \ldots, n$ located around a hypersphere with radius r and center $\boldsymbol{c} \in \mathbb{R}^d$. We assume that the noise samples are mutually independent and distributed according to a multivariate t-distribution with location parameter $\mathbf{0}_d$, scale matrix $\sigma^2 \boldsymbol{I}_d$ with σ^2 unknown, and number of degrees of freedom ν, denoted as $t_d(\mathbf{0}, \sigma^2, \nu)$. We introduce latent vectors $\boldsymbol{u}_i \in \mathcal{S}^{d-1}$, $i = 1, \ldots, n$, where $\mathcal{S}^{d-1}$ is the unit sphere of dimension d. These vectors are assigned a von Mises-Fisher prior with known parameters $\kappa \geq 0$ and $\boldsymbol{\mu} \in \mathbb{R}^d$, denoted as $vMF(\kappa, \boldsymbol{\mu})$[1]. The corresponding

[1]In the practical LIDAR application mentioned before, $\boldsymbol{\mu}$ is the known direction of the LIDAR and ν is a parameter fixed by the user to account for the presence of outiers

model for hypersphere fitting is:

$$\boldsymbol{z}_i = \boldsymbol{c} + r\boldsymbol{u}_i + \boldsymbol{e}_i,\ \boldsymbol{u}_i \sim vMF(\kappa, \boldsymbol{\mu}), \tag{1}$$

$$\boldsymbol{e}_i \sim t_d(\boldsymbol{0}_d, \sigma^2 \boldsymbol{I}_d, \nu). \tag{2}$$

The probability density function (pdf) of a von Mises-Fisher distribution with parameters κ and $\boldsymbol{\mu}$ is given by:

$$f_d(\boldsymbol{x}; \boldsymbol{\mu}, \kappa) = C_d(\kappa) \exp\left(\kappa \boldsymbol{\mu}^\top \boldsymbol{x}\right), \tag{3}$$

with $\kappa \geq 0$, $\|\boldsymbol{\mu}\|_2 = 1$ and $C_d(\kappa)$ is a normalization constant defined by:

$$C_d(\kappa) = \frac{\kappa^{d/2-1}}{(2\pi)^{d/2} I_{d/2-1}(\kappa)}, \tag{4}$$

where I_m is the modified Bessel function of the first kind of order m. Regarding the noise distribution, ν is a parameter fixed by the user in order to control the robustness of the model [13]. Any multivariate t-distribution being a scale mixture of Gaussians [13], the problem can be reformulated as follows:

$$\boldsymbol{z}_i = \boldsymbol{c} + r\boldsymbol{u}_i + \frac{\boldsymbol{y}_i}{\sqrt{\frac{w_i}{\nu}}}, \tag{5}$$

with

$$\boldsymbol{y}_i \sim \mathcal{N}(\boldsymbol{0}_d, \sigma^2 \boldsymbol{I}_d), \quad w_i \sim \chi^2(\nu) = \Gamma\left(\frac{\nu}{2}, \frac{1}{2}\right). \tag{6}$$

The latent variables $\boldsymbol{y}_i$ have a multivariate normal distribution with mean vector $\boldsymbol{0}_d$ and covariance matrix $\sigma^2 \boldsymbol{I}_d$, denoted as $\mathcal{N}(\boldsymbol{0}_d, \sigma^2 \boldsymbol{I}_d)$, whose pdf is

$$f_{\mathcal{N}}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{\exp\left[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right]}{(2\pi)^{d/2} \det(\boldsymbol{\Sigma})^{1/2}}, \tag{7}$$

with $\boldsymbol{\mu} = \boldsymbol{0}_d$ and $\boldsymbol{\Sigma} = \sigma^2 \boldsymbol{I}_d$. The latent variables w_i have a gamma distribution with shape parameter $\frac{\nu}{2}$ and rate parameter $\frac{1}{2}$, denoted as $\Gamma\left(\frac{\nu}{2}, \frac{1}{2}\right)$. The pdf of a gamma distribution with shape parameter α and rate parameter β is:

$$f_\Gamma(x; \alpha, \beta) = x^{\alpha-1} \frac{\beta^\alpha \exp(-\beta x)}{\Gamma(\alpha)} \mathbb{I}_{]0,\infty[}(x), \tag{8}$$

where $\mathbb{I}_A$ is the indicator function on the set A defined by

$$\mathbb{I}_A(x) = \begin{cases} 1 & \text{if } x \in A, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

The latent variables $\boldsymbol{u}_i$ and w_i are supposed to be independent. Thus, the joint distribution of the observations and latent variables conditioned on the model parameters, known as the complete likelihood, can be expressed as:

$$p(\boldsymbol{z}_i, \boldsymbol{u}_i, w_i \mid \boldsymbol{\theta}, \boldsymbol{\psi}) = p(\boldsymbol{z}_i \mid \boldsymbol{u}_i, w_i, \boldsymbol{\theta}, \boldsymbol{\psi}) \times p(w_i \mid \nu) f_d(\boldsymbol{u}_i \mid \kappa, \boldsymbol{\mu}), \tag{10}$$

where $p(w_i \mid \nu) = f_\Gamma\left(w_i; \frac{\nu}{2}, \frac{1}{2}\right)$, $\boldsymbol{\theta} = \{r, \boldsymbol{c}, \sigma^2\}$ is the vector of unknown model parameters and $\boldsymbol{\psi} = \{\kappa, \boldsymbol{\mu}, \nu\}$ contains the model hyperparameters.

B. Likelihood

The likelihood of the observations can be expressed as:

$$\mathcal{L}(\boldsymbol{\theta}; \boldsymbol{Z}, \boldsymbol{\psi}) = \prod_{i=1}^{n} p(\boldsymbol{z}_i \mid \boldsymbol{\theta}, \boldsymbol{\psi}), \\ = \prod_{i=1}^{n} \int_{\mathcal{S}^{d-1}} \int_{\mathbb{R}_+} p(\boldsymbol{z}_i, \boldsymbol{u}_i, w_i \mid \boldsymbol{\theta}, \boldsymbol{\psi}) \, \mathrm{d}w_i \mathrm{d}\boldsymbol{u}_i.$$

The maximum likelihood estimator of the unknown parameter vector $\boldsymbol{\theta}$ maximizing $\mathcal{L}(\boldsymbol{\theta}; \boldsymbol{Z}, \boldsymbol{\psi})$ cannot be expressed in closed-form and cannot be easily computed using a numerical optimization method. Instead, we propose to investigate a Markov chain Monte Carlo (MCMC) method to generate samples asymptotically distributed according to the following augmented posterior distribution

$$p(\boldsymbol{\theta}, \boldsymbol{U}, \boldsymbol{w} \mid \boldsymbol{Z}, \boldsymbol{\psi}) \propto p(\boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{w} \mid \boldsymbol{\theta}, \boldsymbol{\psi}) \, p(\boldsymbol{\theta} \mid \boldsymbol{\psi}), \tag{11}$$

with $\boldsymbol{Z} = [\boldsymbol{z}_1, ..., \boldsymbol{z}_n], \boldsymbol{U} = [\boldsymbol{u}_1, ..., \boldsymbol{u}_n], \boldsymbol{w} = [w_1, ..., w_n]^T$ and $p(\boldsymbol{\theta} \mid \boldsymbol{\psi})$ is the prior distribution of the model parameters. The generated samples are then used to approximate the Bayesian estimators of the model parameters.

C. Prior distributions

The hyperparameters r and $\boldsymbol{c}$ are assigned independent non informative uniform improper distributions, i.e.,

$$f_r(r) = \mathbb{I}_{\mathbb{R}_+}(r), f_c(\boldsymbol{c}) = \mathbb{I}_{\mathbb{R}^d}(\boldsymbol{c}). \tag{12}$$

The prior distribution for σ^2 is a non-informative conjugate Jeffrey's prior [14], which reflects the absence of knowledge regarding this parameter and will simplify the analysis:

$$f_{\sigma^2}(\sigma^2) = \frac{1}{\sigma^2} \mathbb{I}_{\mathbb{R}_+}(\sigma^2). \tag{13}$$

D. Conditional distributions of the augmented posterior

This section derives the conditional distributions of (10) that are required to implement the Gibbs sampler. Using (3), (7), (8) and (10), the posterior distribution of the unknown and latent parameters can be written as:

$$p(\boldsymbol{z}_i, \boldsymbol{u}_i, w_i \mid \boldsymbol{\theta}, \boldsymbol{\psi}) = \left(\frac{\nu 2\pi\sigma^2}{w_i}\right)^{-\frac{d}{2}} \exp\left(-\frac{w_i}{2\nu\sigma^2} \|\boldsymbol{z}_i - \boldsymbol{c} - r\boldsymbol{u}_i\|_2^2\right) \times C_d(\kappa) \exp(\kappa \boldsymbol{\mu}^\top \boldsymbol{u}_i) \frac{(w_i)^{\frac{\nu}{2}-1}}{\Gamma\left(\frac{\nu}{2}\right) 2^{\frac{\nu}{2}}} \exp\left(\frac{-w_i}{2}\right). \tag{14}$$

According to Bayes' theorem,

$$p(\boldsymbol{u}_i, w_i \mid \boldsymbol{z}_i, \boldsymbol{\theta}, \boldsymbol{\psi}) \propto p(\boldsymbol{z}_i, \boldsymbol{u}_i, w_i \mid \boldsymbol{\theta}, \boldsymbol{\psi}), \tag{15}$$

where $\propto$ means "proportional to", which leads to:

$$p(\boldsymbol{u}_i, w_i \mid \boldsymbol{z}_i, \boldsymbol{\theta}, \boldsymbol{\psi}) \propto w_i^{\alpha-1} \exp(-w_i \beta_i) \exp(\kappa_i \boldsymbol{\mu}_i^\top \boldsymbol{u}_i), \tag{16}$$

with the following notations

$$\boldsymbol{\mu}_i = \frac{w_i r(\boldsymbol{z}_i - \boldsymbol{c}) + \nu\sigma^2\kappa\boldsymbol{\mu}}{\|w_i r(\boldsymbol{z}_i - \boldsymbol{c}) + \nu\sigma^2\kappa\boldsymbol{\mu}\|_2}, \tag{17}$$

$$\kappa_i = \frac{\|w_i r(\boldsymbol{z}_i - \boldsymbol{c}) + \nu\sigma^2\kappa\boldsymbol{\mu}\|_2}{\nu\sigma^2}, \tag{18}$$

$$\alpha = \frac{d+\nu}{2}, \tag{19}$$

$$\beta_i = \frac{1}{2}\left(\frac{\|\boldsymbol{z}_i - \boldsymbol{c}\|_2^2 + r^2}{\nu\sigma^2} + 1\right). \tag{20}$$

The conditional distributions of $\boldsymbol{u}_i$ and w_i can be determined from (16) leading to:

$$\boldsymbol{u}_i \mid \boldsymbol{z}_i, w_i, \boldsymbol{\theta}, \boldsymbol{\psi} \sim vMF(\kappa_i, \boldsymbol{\mu}_i), \tag{21}$$

and

$$w_i \mid \boldsymbol{z}_i, \boldsymbol{u}_i, \boldsymbol{\theta}, \boldsymbol{\psi} \sim \Gamma(\alpha, \beta_i). \tag{22}$$

Similarly, by denoting $\boldsymbol{\theta}_r = \boldsymbol{\theta} \setminus \{r\}$, the conditional distribution of r can be determined:

$$p(r \mid \boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{w}, \boldsymbol{\theta}_r, \boldsymbol{\psi}) \propto p_r(r) \prod_{i=1}^{n} p(\boldsymbol{z}_i, \boldsymbol{u}_i, w_i \mid \boldsymbol{\theta}, \boldsymbol{\psi}). \tag{23}$$

Using (12), (14) and (23), the following result is obtained:

$$r \mid \boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{w}, \boldsymbol{\theta}_r, \boldsymbol{\psi} \sim \mathcal{N}_{]0,\infty[}(\mu_r, \sigma_r^2), \tag{24}$$

where $\mathcal{N}_{]0,\infty[}(\mu_r, \sigma_r^2)$ is the Gaussian distribution with mean μ_r and variance σ_r^2 truncated on $]0, \infty[$ and

$$\mu_r = \frac{\sum_{i=1}^{n} w_i \boldsymbol{u}_i^\top (\boldsymbol{z}_i - \boldsymbol{c})}{\sum_{i=1}^{n} w_i}, \sigma_r^2 = \frac{\sigma\sqrt{\nu}}{\sqrt{\sum_{i=1}^{n} w_i}}. \tag{25}$$

Similarly, by denoting $\boldsymbol{\theta}_c = \boldsymbol{\theta} \setminus \{\boldsymbol{c}\}$, the conditional distribution of c can be determined:

$$p(\boldsymbol{c} \mid \boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{w}, \boldsymbol{\theta}_c, \boldsymbol{\psi}) \propto p_c(\boldsymbol{c}) \prod_{i=1}^{n} p(\boldsymbol{z}_i, \boldsymbol{u}_i, w_i \mid \boldsymbol{\theta}, \boldsymbol{\psi}), \tag{26}$$

which leads to:

$$\boldsymbol{c} \mid \boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{w}, \boldsymbol{\theta}_c, \boldsymbol{\psi} \sim \mathcal{N}(\boldsymbol{\mu}_c, \sigma_c^2 \boldsymbol{I}_d), \tag{27}$$

with

$$\boldsymbol{\mu}_c = \frac{\sum_{i=1}^{n} w_i (\boldsymbol{z}_i - r\boldsymbol{u}_i)}{\sum_{i=1}^{n} w_i}, \sigma_c^2 = \frac{\sigma\sqrt{\nu}}{\sqrt{\sum_{i=1}^{n} w_i}}. \tag{28}$$

Finally, denoting $\boldsymbol{\theta}_\sigma = \boldsymbol{\theta} \setminus \{\sigma^2\}$, the conditional distribution of σ^2 can be computed:

$$p(\sigma^2 \mid \boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{w}, \boldsymbol{\theta}_\sigma, \boldsymbol{\psi}) \propto p_{\sigma^2}(\sigma^2) \prod_{i=1}^{n} p(\boldsymbol{z}_i, \boldsymbol{u}_i, w_i \mid \boldsymbol{\theta}, \boldsymbol{\psi}). \tag{29}$$

Using (13), (14) and (29), the following result is obtained:

$$\sigma^2 \mid \boldsymbol{Z}, \boldsymbol{U}, \boldsymbol{w}, \boldsymbol{\theta}_\sigma, \boldsymbol{\psi} \sim \mathcal{IG}(\alpha_{\sigma^2}, \beta_{\sigma^2}) \tag{30}$$

with

$$\alpha_{\sigma^2} = \frac{nd}{2},\ \beta_{\sigma^2} = \frac{1}{2\nu} \sum_{i=1}^{n} w_i \|\boldsymbol{z}_i - \boldsymbol{c} - r\boldsymbol{u}_i\|_2^2, \tag{31}$$

where $\mathcal{IG}(\alpha, \beta)$ is the inverse gamma distribution with shape parameter α and scale parameter β, whose pdf is:

$$f_{\mathcal{IG}}(x; \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{x}\right)^{\alpha+1} \exp\left(-\frac{\beta}{x}\right) \mathbb{I}_{\mathbb{R}^+}(x). \tag{32}$$

III. Gibbs Sampler

The Gibbs sampler [15] is an MCMC method generating samples of each parameter of the posterior distribution according to its conditional distribution. Since the distribution of the generated samples converges to the posterior distribution, their sample mean after a *burn-in* period is a good approximation of the MMSE estimators. The Gibbs sampler for the hypersphere fitting problem is summarized in Alg. 1. It generates n_{samples} of the different parameters, discards the $n_{\text{burn-in}}$ first samples (to eliminate the samples too dependent on the initialization) and uses the remaining samples to build the parameter estimates.

Algorithm 1 Gibbs sampler

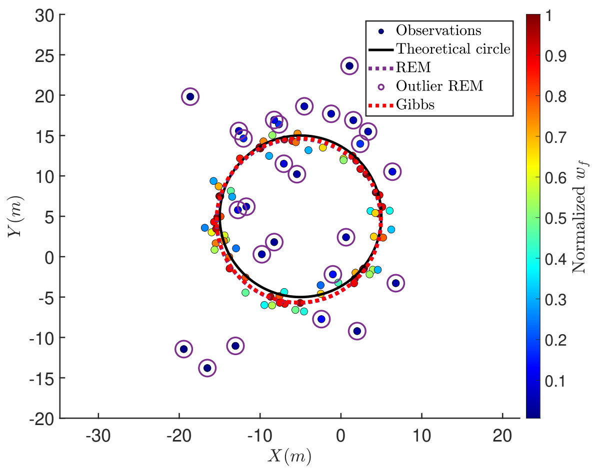
Input: $\boldsymbol{Z}$, $\boldsymbol{\psi}$, n_{samples}, $n_{\text{burn-in}}$
Output: $\boldsymbol{c}_f$, r_f, σ_f^2

```
u(0) ← u_0
w(0) ← w_0
r(0) ← r_0
c(0) ← c_0
σ²(0) ← σ²_0
for t = 1, ..., n_samples do
    U(t) ← p(U | w(t−1), Z, θ(t−1), ψ) ▷ (21)
    w(t) ← p(w | U(t), Z, θ(t−1), ψ) ▷ (22)
    r(t) ← p(r | U(t), w(t), Z, c(t−1), σ²(t−1), ψ) ▷ (24)
    c(t) ← p(c | U(t), w(t), Z, r(t), σ²(t−1), ψ) ▷ (27)
    σ²(t) ← p(σ² | U(t), w(t), Z, r(t), c(t), ψ) ▷ (30)
end for
r_f ← mean(r(n_burn-in: end))
c_f ← mean(c(n_burn-in: end))
σ²_f ← mean(σ²(n_burn-in: end))
w_f ← mean(w(n_burn-in: end))
```

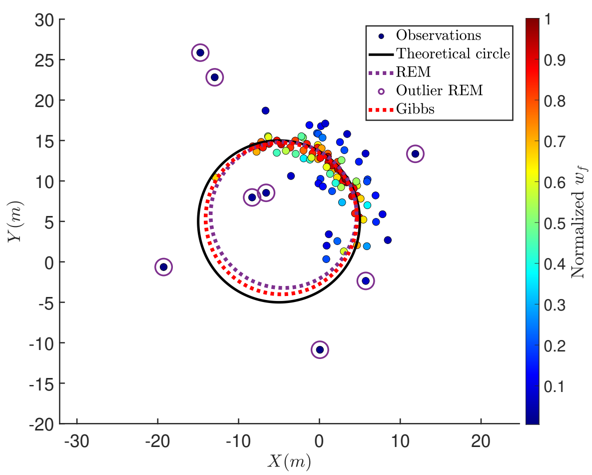
IV. Simulation Results

Several experiments have been performed in order to evaluate the performance of the proposed estimation method. This method requires three parameters to be adjusted: the total number of iterations (i.e., the total number of draws), the number of initial draws used for *burn in*, and the number of degrees of freedoms ν. The two first parameters are chosen in order to ensure the algorithm convergence. This convergence can be assessed using the so-called potential scale reduction factor $\sqrt{\hat{\rho}}$ (a value of $\sqrt{\hat{\rho}}$ below 1.2 is recommended in [16, p. 332]). In the following experiments, the center of the hypersphere was initialized to the mean of the noisy measurements denoted by $\boldsymbol{c}_0$, and its radius was initially set to its MLE given the center $\boldsymbol{c}_0$, i.e., $r_0 = \frac{1}{n}\sum_{i=1}^{n}\|\boldsymbol{z}_i - \boldsymbol{c}_0\|_2$. The noise variance was initialized by its MLE given $(\boldsymbol{c}_0, r_0)$, i.e., $\sigma_0^2 = \frac{1}{nd}\sum_{i=1}^{n}\|\boldsymbol{z}_i - \boldsymbol{c}_0\|_2^2 - \frac{1}{d}r_0^2$. The latent variables were initially set to $\boldsymbol{u}_i = \boldsymbol{\mu}$ and $w_i = 1$, $\forall i = 1, \ldots, n$. Finally, we place ourselves in an ideal case where the value of ν, the hyperparameter of the Gibbs agorithm, is equal to the value of ν_g, the degree of freedom used to generate the data.

A. Synthetic 2D Dataset



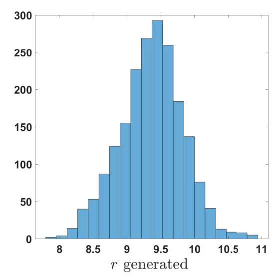
(a) Case n°1: Uniform distribution on the sphere.



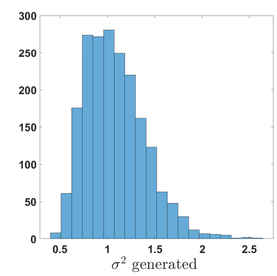
(b) Case n°2: Directional distribution on a part of the sphere.

Fig. 1. Comparison between REM and Gibbs algorithms. The first line corresponds to data generated with $\kappa = 0$. The second line corresponds to data generated with $\kappa = 3$.

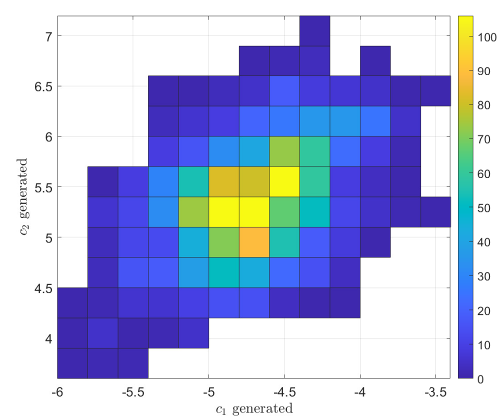
This section compares the proposed method referred to as Gibbs with the REM method presented in [12] for observations generated with $\nu_g = 1$. Experiments are conducted with data generated with $n = 100$, $d = 2$, $r = 10$, $\boldsymbol{c} = (-5, 5)^T$ and $\sigma = 1$. The first experiments are carried out with a uniform distribution on the sphere (i.e., $\kappa = 0$) whereas the other experiments use a distribution concentrated on a part of the sphere (with $\kappa = 3$ and $\boldsymbol{\mu} = (1,1)/\sqrt{2}$). The field of view of the sensor is supposed to be FoV $= [-30, 20; -20, 30]$, i.e., all measurements falling outside this region of interest are discarded. The total number of samples for the Gibbs sampler is $n_\mathrm{t} = 5000$ with $n_\mathrm{bi} = 3000$ *burn in* iterations (i.e., $n_\mathrm{mc} = 2000$ samples are used to build the estimates). The number of iterations used for the REM algorithm is 60. Figure 1 shows the generated samples (referred to as "observations") (note the presence of outliers due to the t distribution), and the estimated circles using the proposed Gibbs sampler and the reference method REM. The proposed Gibbs sampler is clearly more robust to outliers than REM for directional data. In the case of a uniform distribution on the circle, both algorithms tend towards similar solutions. This figure also highlights the way the two algorithms operate: REM estimates the parameters of the hypersphere by indicating that an observation is an outlier using a Bernoulli distribution. Conversely, the Gibbs sampler estimates these parameters by assigning a weight to each observation resulting from the Student-t distribution assigned to the noise. Figure 2 shows the histogram of all parameters of interest $(r, \boldsymbol{c} = (c_1, c_2), \sigma^2)$ generated for the estimation (i.e., after the *burn in*) in a case of directional data. These histograms are in good agreement with the marginal distribution of each variable of interest.



(a) Histogram of r



(b) Histogram of σ^2



(c) Histogram of $\boldsymbol{c}$

Fig. 2. Histograms of estimated parameters.

B. Monte-Carlo simulations

Monte-Carlo simulations allow the performance of the proposed algorithm to be quantified. The data were generated with the parameters indicated in Section IV-A for directional data. A first experiment compares the robustness of the proposed

Gibbs sampler for different values of ν_g with that of REM. Fig. 3 shows that the proposed Gibbs sampler provides better performance than REM for all values of ν_g, in the sense that the MSEs of the radius and center estimates obtained using the Gibbs sampler are lower than those using REM. The robustness of the proposed algorithm to additive noise can be evaluated using the results displayed in Fig. 4 (data generated using $\nu_g = 2$) showing better performance for the Gibbs sampler than for REM. Finally the execution times of the two algorithms are reported in Table I showing the price to pay with MCMC methods.

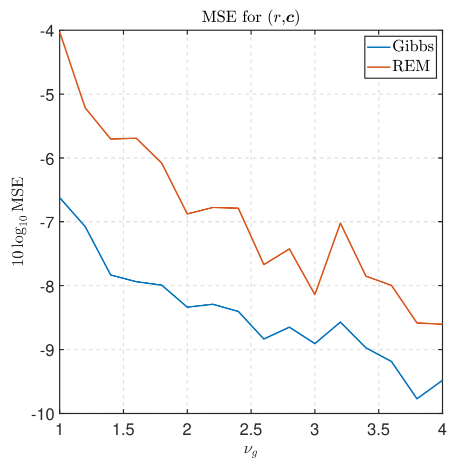


Fig. 3. MSEs of the estimates of $(r, \boldsymbol{c})$ versus ν_g for Gibbs and REM.

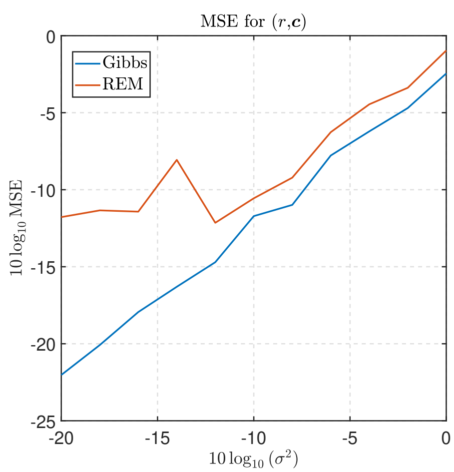


Fig. 4. MSEs of the estimates of $(r, \boldsymbol{c})$ versus σ^2 for Gibbs and REM.

TABLE I
COMPUTATION TIME

Algorithm	**Time for 100 Points (s)**	**Time for 500 Points (s)**	**Time for 1000 Points (s)**
Gibbs	8.5×10^{-1}	3.9	9.1
REM	4.1×10^{-3}	9.9×10^{-3}	1.5×10^{-2}

V. CONCLUSION

This paper proposed a robust Bayesian sampling algorithm for hypersphere fitting. A student t-distribution was assigned to the noise samples in order to mitigate the presence of outliers whereas a von Mises-Fisher distribution was assigned to latent variables in order to exploit the fact that observations are located in a part of the hypersphere. The proposed algorithm requires the knowledge of three parameters: the number of generated samples, the number of *burn in* iterations, and the number of degrees of freedom associated with the t-distribution. The results obtained with the proposed Gibbs sampler are very encouraging compared to the REM algorithm of [12]. Future work includes the estimation of the parameters of the von Mises-Fisher distribution within the MCMC framework and the generalization of the algorithm to multiple hyperspheres for an application to real LiDAR data.

REFERENCES

[1] F. Baret *et al.*, "Phenomobile: A fully automatic robot for high-throughput field phenotyping of a large range of crops with active measurements," in *Proc. Workshop Image Analysis Methods in the Plant Sciences (IAMPS)*, (Lyon, France), Jul. 2019.
[2] J. Lesouple, B. Pilastre, Y. Altmann, and J.-Y. Tourneret, "Hypersphere fitting from noisy data using an EM algorithm," *IEEE Signal Process. Lett.*, vol. 28, pp. 314–318, Jan. 2021.
[3] D. Epstein and D. Feldman, "Sphere fitting with applications to machine tracking," *Algorithms*, vol. 13, Jul. 2020.
[4] B. Liu, R. Tharmarasa, R. Jassemi, D. Brown, and T. Kirubarajan, "Extended target tracking with multipath detections, terrain-constrained motion model and clutter," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, pp. 7056–7072, Nov. 2021.
[5] N. Wahlström and E. Özkan, "Extended target tracking using Gaussian processes," *IEEE Trans. Signal Process.*, vol. 63, no. 16, pp. 4165–4178, 2015.
[6] P. Nunez, R. Vazquez-Martin, A. Bandera, and F. Sandoval, "An algorithm for fitting 2-D data on the circle: Applications to mobile robotics," *IEEE Signal Process. Lett.*, vol. 15, pp. 127–130, 2008.
[7] D. Epstein and D. Feldman, "Quadcopter tracks quadcopter via real-time shape fitting," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 544–550, 2018.
[8] F. Bonin-Font, A. Ortiz, and G. Oliver, "Visual navigation for mobile robots: A survey," *J. Intell. Robotics Syst.*, vol. 53, p. 263–296, May 2008.
[9] D.-T. Lin and C.-M. Yang, "Real-time eye detection using face-circle fitting and dark-pixel filtering," in *Proc. Int. Conf. Multimedia and Expo (ICME)*, (Taipei, Taiwan), June 2004.
[10] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, (Providence, RI, USA), June 2012.
[11] L. Pan, W.-S. Chu, J. M. Saragih, F. De la Torre, and M. Xie, "Fast and robust circular object detection with probabilistic pairwise voting," *IEEE Signal Process. Lett.*, vol. 18, no. 11, pp. 639–642, 2011.
[12] J. Lesouple, B. Pilastre, Y. Altmann, and J.-Y. Tourneret, "Robust hypersphere fitting from noisy data using an EM algorithm," in *Proc. European Signal Processing Conference (EUSIPCO)*, (Dublin, Ireland), pp. 1–5, Aug. 2021.
[13] F. Z. Doğru, Y. M. Bulut, and O. Arslan, "Doubly reweighted estimators for the parameters of the multivariate t-distribution," *Communications in Statistics - Theory and Methods*, vol. 47, pp. 4751–4771, Feb. 2018.
[14] H. Jeffreys, "An invariant form for the prior probability in estimation problems," *Proc. R. Soc. Lond. A*, vol. 186, pp. 453–461, Sep. 1946.
[15] C. M. Bishop, "Sampling methods," in *Pattern Recognition and Machine Learning (Information Science and Statistics)*, ch. 11, pp. 523–559, Berlin, Heidelberg: Springer-Verlag, 2006.
[16] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian data analysis*. Chapman and Hall/CRC, 1995.