Optimal Transport : an application to the RADAR Recognition Process for deinterleaving RADAR pulses and identifying emitter

Manon Mottier

To cite this version:

Manon Mottier. Optimal Transport : an application to the RADAR Recognition Process for deinterleaving RADAR pulses and identifying emitter. Signal and Image Processing. Université Paris-Saclay, 2024. English. NNT : 2024UPAST070 . tel-04653381

HAL Id: tel-04653381

https://theses.hal.science/tel-04653381

Submitted on 18 Jul 2024

Optimal Transport: an application to the RADAR Recognition Process for deinterleaving RADAR pulses and identifying emitters

Transport Optimal : une application au Processus de Reconnaissance RADAR pour désentrelacer les impulsions RADAR et identifier les émetteurs

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 580, sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat: Informatique mathématique
Graduate School : Sciences de l'ingénierie et des systèmes.
Référent : CentraleSupélec

Thèse préparée dans l'unité de recherche du Laboratoire des signaux et systèmes (Université Paris-Saclay, CNRS, CentraleSupélec),
sous la direction de Frédéric PASCAL, Professeur à CentraleSupélec,
la co-direction de Gilles CHARDON, Maître de conférence à CentraleSupélec,
et la co-supervision de Jean-Daniel BUSI, Architecte Système chez Avantix.

Thèse soutenue à Paris-Saclay, le 28 juin 2024, par

Manon MOTTIER

THESE DE DOCTORAT

NNT : 2024UPAST070

Composition du jury

Membres du jury avec voix délibérative

Audrey GIREMUS Professeure des universités, Université de Bordeaux	Présidente
Rémi FLAMARY Professeur École Polytechnique, Institut Polytechnique de Paris	Rapporteur
Ali KHENCHAF Professeur des Universités, ENSTA Bretagne	Rapporteur
Jean-Philippe OVARLEZ Directeur de recherche, ONERA Saclay	Examinateur

université
PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies
de l'information et de
la communication (STIC)

Titre: Processus de Reconnaissance RADAR basé sur le transport optimal pour désentrelacer un signal RADAR et identifier les émetteurs.

Mots clés : Processus de Reconnaissance RADAR, Désentrelacement, Identification, Transport optimal, Guerre électronique, Clustering, Radar passif

Résumé : Le renseignement militaire est un aspect essentiel pour la sécurité et la défense d'un pays, notamment le renseignement d'origine électromagnétique (ROEM). L'émergence des systèmes passif a permis de donner un avantage considérable aux acteurs capables de les maitriser en permettant une surveillance discrète et à moindre coût. Néanmoins, l'interception et le traitement des signaux par un RADAR passif nécessitent la mise en place d'une chaine de traitement algorithmique dédiée, capable de comprendre la diversité des spectres électromagmétiques ainsi que les phénomènes physiques sous-jacents. Au fil des années, les enjeux se sont complexifiés et diversifiés notamment à cause de nombreuses innovations technologiques qui ont conduit à la complexification et de la sophistication des équipements électroniques ; les RADARs ont des spectres électromagnétiques plus similaires rendant leur différenciation complexe. Ces travaux proposent un Processus de Reconnaissance RADAR permettant dans un premier temps de désentrelacer un signal puis d'identifier les RADARs. Tout d'abord deux nouvelles approches de désentrelacement non supervisées sont proposées, basées sur une combinaison d'algorithmes de clustering intégrant des distances de transport optimal afin de séparer les impulsions en plusieurs clusters avant de les regrouper les clusters appartenant à un même RADAR. Enfin, lorsque la phase de désentrelacement est terminée, l'identification des RADARs est faite à partir de l'élaboration d'une distance de transport optimal entre une base de données de référence et les ensembles d'impulsions précédemment désentrelacés tout en modélisant le phénomène d'impulsions manquantes.

Title: RADAR Recognition Process based on optimal transport to deinterleave a RADAR signal and identify emitters.

Keywords: RADAR Recognition Process, Deinterleaving, Identification, Optimal transport, Electronic warfare, Clustering, Passive Radar

Abstract: Military intelligence is essential to a country's security and defense, particularly signals intelligence (ROEM). The emergence of passive systems has given a considerable advantage to those capable of controlling them by allowing discreet surveillance at a lower cost. However, the interception and processing of signals by a passive RADAR require establishing a dedicated algorithmic processing chain capable of understanding the diversity of electromagnetic spectra and the underlying physical phenomena. Over the years, the issues have become more complex and diversified, mainly because of numerous technological innovations that have led to the complexity and sophistication of electronic equipment; RADARs have more similar electromagnetic spectra, making their differentiation complex. This work proposes a RADAR Recognition Process first to deinterleave a signal and then to identify the RADARs. First, two new unsupervised deinterleaving approaches are proposed based on a combination of clustering algorithms integrating optimal transport distances to separate the pulses into several clusters before grouping the clusters belonging to the same RADAR. Finally, when the deinterleaving phase is completed, the RADARs are identified by developing an optimal transport distance between a reference database and the sets of previously deinterleaved pulses while modeling the phenomenon of missing pulses.

Funding and supervision

The scientific productions were part of a CIFRE thesis in collaboration with the Signals and System Laboratory (L2S) of CentraleSupélec at Paris-Saclay University in France and Avantix, a French company. Frédéric PASCAL and Gilles CHARDON from the L2S provided their scientific expertise and an environment conducive to technological innovation.

Avantix, a subsidiary of the ATOS group, funded all of this research and provided all of its RADAR expertise and data, mainly thanks to the supervision of Jean-Daniel BUSI and Chakib BELAFDIL. Avantix is a French company specializing in electronic warfare. Avantix designs critical high-tech systems to provide homeland protection and military forces, specifically in the Radar processing chain.

The Direction Générale de l'Armement (DGA) also provided its expertise and data to evaluate the work developed within the thesis framework.

Contents

*In the darkest times, perseverance
and resilience are the lights that
guide us to flourishing.*

Acknowledgement

Je tenais à exprimer ma profonde gratitude et mes sincères remerciements pour la collaboration exceptionnelle dont j'ai profité au sein du laboratoire des Signaux et Système de CentraleSupélec ainsi qu'Avantix. J'ai passé des années à étudier dans un cadre privilégié, accompagnée par des personnes avec qui cela a été un réel de plaisir de travailler. Cette collaboration a été une expérience enrichissante, autant d'un point de vue professionnel que personnel et je suis extrêmement reconnaissante d'avoir eu cette opportunité de pouvoir travailler sur ce sujet et avec cette équipe encadrante.

J'ai eu la chance d'être entourée durant ces années de thèse de ma famille, d'amis et de collègues qui ont très largement contribué à mon épanouissement et à ma réussite. Même en voulant raccourcir ces remerciements, je ne pourrais pas le faire, car certains d'entre vous méritent une attention toute particulière. La longue liste des remerciements peut commencer, en espérant n'oublier personne sur le chemin...

On dit souvent que la réussite d'une thèse dépend principalement de ces encadrants. Pendant ces trois années de vie consacrées à la recherche, on passe beaucoup de temps avec les mêmes personnes, on se remet en question, on traverse des périodes de doute et on accorde une grande importance aux conseils de ses encadrants. Je ne le dirai jamais assez, mais je me sens extrêmement privilégiée d'avoir été entourée par de Frédéric, Gilles ainsi que Jean-Daniel.

Je remercie Frédéric pour son expertise et tous les conseils qu'il m'as apportés durant ce parcours exigeant. Je tiens à exprimer mes plus sincères remerciements à Gilles dont le soutien et l'expertise ont été essentiels pour mener à bien cette thèse. Travailler avec Gilles a été une expérience formidable, non seulement grâce à sa pédagogie, à sa patience, mais aussi grâce aux moments passés ensemble en dehors du cadre académique.

Sans Philippe et Jean-Daniel cette aventure n'aurait pas été possible. Philippe m'a donné l'opportunité de rejoindre Avantix, de travailler sur un sujet que je ne connaissais pas et fais découvrir un secteur d'activité dans lequel je me suis épanoui. L'expertise de Jean-Daniel et son accompagnement tout au long de la thèse m'ont permis de me former et de réaliser cette très belle thèse. Merci également à Chakib pour son écoute et tous les conseils qu'il m'as prodigués.

Au-delà de cette équipe encadrante composée de scientifiques et d'industriels talentueux, et malgré la distance avec le laboratoire, cela a été un réel plaisir de venir (ou d'essayer selon l'humeur du RER B) chaque jour au L2S. J'ai eu l'opportunité de rencontrer de nombreux doctorants et enseignants, avec une mention spéciale pour mon couloir ainsi que pour le GEEPS, avec qui j'ai tissé des liens au fil des années. Je n'oublierai pas les festins partagés au CROUS ni les midis tarot, où je perdais souvent car je ne savais pas compter les cartes (la triche est courante au L2S. . .).

Commençons par Sébastien et Fabien, mes voisins de bureau, avec qui j'ai commencé ma thèse quasiment en même temps. Je repense à toutes les pauses-café que nous avons prises ensemble pour discuter de nos incertitudes, de nos premiers articles publiés, de nos encadrants, ou d'autres sujets. Je vous remercie pour votre soutien constant tout au long de la thèse. Je suis très heureuse d'avoir pu partager ces moments avec vous et de pouvoir continuer à en partager d'autres.

Certaines rencontres m'ont beaucoup fait évoluer et grandir ; je pense notamment à Sarah. Bien que nous n'ayons pas passé beaucoup de temps dans ce bureau en Bréguet, nous avons rapidement partagé des passions communes comme notre engagement pour les droits des femmes, la lutte contre le harcèlement et la discrimination des doctorants ou notre passion pour les plots. Personne ne comprend mieux que toi le temps qu'il faut pour réaliser de belles présentations en LaTeX et la satisfaction qui en découle...

Mais tout ne se résume pas qu'à échanger des conseils sur le doctorat ou sur Gilles. Cette rencontre qui a débuté dans la cadre de la thèse a très vite évolué en une très belle amitié pour laquelle des mots ne suffiront pas à exprimer toute l'affection que je te porte. Je heureuse de te compter parmi mes proches et je suis ravie de pouvoir assister à ton évolution, que ce soit sur le plan personnel que professionnel. Il est rare de rencontrer des personnes telles que toi.

Au cours de ces années, j'ai fait de nombreuses rencontres, dont certaines ont été particulièrement marquantes pour moi. Les afterworks, les conférences et les écoles d'été sont des occasions précieuses pour les doctorants de créer leur réseau professionnel, mais aussi pour rencontrer des personnes avec qui des liens personnels se sont tissés. Je pense à Rebecca, Dora, Pierre, Coraline et Emilien. Je vous remercie pour le soutien que vous m'avez apporté, surtout en cette fin de thèse ; votre présence m'a été d'une grande aide. Je suis ravie de vous avoir eus à mes côtés et de vous avoir découverts sur le plan personnel. J'ai hâte d'assister à vos soutenances. Coraline, sache que depuis que tu reçois les invitations Teams de Pierre, tu es officiellement considérée comme une doctorante. J'attends avec impatience ta soutenance.

Pour finir mes remerciements envers la communauté scientifique, je remercie également les membres du jury pour leurs disponibilités ainsi que pour leur expertise qui a fortement contribué à l'amélioration de mon travail.

Les remerciements continuent et se ressemblent de plus en plus. J'essaie de les personnaliser pour vous exprimer au mieux ma reconnaissance mais ça commence à être difficile. Tout d'abord je remercie ma famille pour avoir été là et d'avoir cru en moi.

Certaines personnes ont pu suivre mon évolution durant cette thèse, à partir du moment où j'ai entamé mes recherches pour faire une thèse jusqu'à la soutenance. Justine j'ai pu partager cette thèse avec toi durant nos années chez Atos, Adrien à travers tous nos appels depuis l'autre bout de l'Île-de-France pour nos pauses cafés et enfin Sarra pendant nos longues soirées valorant & plots ; votre soutien m'a énormément apporté. Je ne vous remercierais jamais assez.

Nous voici enfin à la fin ; merci à mes amis non scientifiques parisiens bien trop nombreux pour vous citer un par un de m'avoir soutenu depuis le début de cette thèse. Vous avez été là à chaque étape de ce parcours et poussé dans les moments difficiles. Je suis extrêmement reconnaissante d'avoir un groupe d'amis aussi formidable et qui m'a soutenu tout au long de ce parcours exigeant. Mention spéciale pour Thomas et Raphaël qui m'ont fait répéter mes premières présentations orales pendant lesquelles j'étais beaucoup trop stressée et avec qui je n'ai cessé de discuter de ma thèse pendant de nombreuses soirées. Ne vous inquiétez pas, d'autres réunions au sommet auront lieu. Enfin merci à Mila qui m'a beaucoup épaulé et à Soraya qui a été d'un soutien indéfectible tout au long de ce parcours semé d'embuches.

Bien que ce diplôme me soit personnellement adressé, il vous en revient une grande part de mérite. Merci à tous ces encadrants de l'ombre.

Resume in French

Le renseignement militaire constitue une préoccupation importante pour la sécurité et la défense d'un pays. Dans un monde en constante évolution, les enjeux liés à la défense et à la sécurité nationales sont complexes et diversifiés, comme la prévention des menaces extérieures, du terrorisme, de la piraterie et de la prolifération des armes de destruction massive, nécessitant la mise en œuvre de méthodes préventives et de stratégies appropriées pour faire face à des menaces en constante évolution. Les informations collectées permettent aux institutions de réagir, de gérer et d'anticiper rapidement les crises nationales et internationales.

Ce défi est particulièrement apparent dans le renseignement électronique (ELINT) [Wil06; Sko08; Sch80], où l'accent est mis sur la collecte et l'utilisation de signaux électromagnétiques provenant de diverses sources pour extraire des informations sensibles afin de guider les opérateurs du renseignement électronique (ESM) dans leurs engagements stratégiques. Les systèmes RADAR sont fréquemment utilisés pour détecter les menaces et mettre en œuvre des contre-mesures appropriées, en particulier avec des systèmes passifs offrant des approches discrètes d'écoute et de collecte de données à moindre coût.

La détection des émissions RADAR est essentielle pour identifier les menaces, surveiller les mouvements ennemis, protéger les forces amies et fournir une stratégie défensive. Cependant, les défis sont nombreux, principalement en raison de la diversité des signaux, ou encore des techniques de modulation sophistiquées. Les développements technologiques récents ont conduit à de nombreuses innovations dans le domaine de la défense en raison de la complexité et de la sophistication croissantes des équipements électroniques ; Les émetteurs RADAR ont des spectres électromagnétiques plus similaires, ce qui rend leur différenciation complexe. Ce défi est d'autant plus vrai que les signaux interceptés sont volumineux et contiennent davantage d'informations à traiter.

L'intelligence artificielle (IA) permet aux machines d'adapter, de développer et d'exploiter leurs algorithmes rapidement, offrant un avantage significatif aux organisations capables de maîtriser les outils et de transformer la chaîne de traitement du signal RADAR. Tous ces changements représentent un défi continu et nécessitent une expertise technique avancée, une compréhension approfondie des systèmes RADAR et une adaptation constante aux développements technologiques pour fournir de nouvelles techniques plus précises pour séparer les impulsions reçues et identifier les émetteurs

RADAR d'un signal intercepté.

Ces recherches visent à répondre à un besoin croissant pour développer des méthodes avancées d'apprentissage automatique pour l'intelligence du RADAR passif, plus particulièrement pour améliorer le processus de reconnaissance RADAR grâce au transport optimales. Le processus de reconnaissance RADAR peut être défini en deux étapes : la première étape consiste à désentrelacer un signal en séparant et en regroupant les impulsions mélangées d'un nombre inconnu d'émetteurs avant de les identifier dans un second temps.

Dans la suite, l'accent est mis sur le traitement des signaux interceptés par des capteurs passifs, qui, par abus de langage, sont appelés émetteurs. Les données des émetteurs RADAR sont cruciales dans de nombreux domaines, tels que la reconnaissance d'objets, la détection d'anomalies et la prise de décision autonome. La collecte de données d'émetteurs RADAR réels peut être coûteuse et limitée en quantité, ce qui rend les données RADAR simulés extrêmement précieuses. Les données simulées doivent représenter avec précision les caractéristiques du système des émetteurs RADAR. Des méthodes avancées de modélisation et de simulation sont nécessaires pour reproduire de manière réaliste les signaux des émetteurs RADAR, en tenant compte des effets du bruit, des interférences et des caractéristiques environnementales. Les méthodologies proposées ont été développées et évaluées à l'aide de plusieurs simulateurs de données : un simulateur confidentiel fourni par Avantix, et un second développé au cours des recherches, qui inclut des détails plus complets. Les signaux utilisés sont simulés à partir des caractéristiques des classes d'émetteurs, structurées pour faciliter leur compréhension et leur utilisation à travers les approches.

Le désentrelacement d'un signal est une étape critique dans la chaîne de traitement RADAR. Les opérateurs ESM ont besoin d'informations pertinentes pour analyser un signal intercepté et extraire les informations significatives pour séparer les impulsions des émetteurs dans le signal sans information préalable; ces conclusions peuvent avoir des conséquences dramatiques selon les circonstances. A partir d'un signal reçu contenant des impulsions provenant d'un nombre inconnu d'émetteurs, deux nouvelles approches simples et non supervisées basées sur une combinaison d'algorithmes de clustering et de distances de transport optimales sont introduites pour désentrelacer un signal.

La première approche, nommée HACOT, est basée sur une stratégie en deux étapes combinant l'algorithme de clustering non supervisé HDSBCAN [CMS13] ("the unsupervised hierarchical density-based spatial clustering of applications with noise") avec un clustering agglomératif hiérarchique (HAC) [Joh67; CPD20] intégrant des distances de transport optimales [Vil09; Bon+11]. La première étape consiste à séparer les impulsions avec HDBSCAN à partir de la fréquence et de la durée d'impulsion sous contrainte que les impulsions d'émetteurs différents ne soient pas regroupées dans le même cluster. Ensuite, comme les émetteurs présentent des caractéristiques complexes et peuvent être représentés par plusieurs clusters, un clustering agglomératif hiérarchique basé sur les distances de transport optimale est appliqué pour fusionner ces clusters. La modernisation des équipements technologiques et le bruit peuvent fortement corrompre la qualité des signaux interceptés étalant les impulsions dans le plan à partir duquel le clustering a été effectué rendant ce clustering inefficace. De plus, lorsque le périmètre d'écoute concerne certaines zones qui sont composées de RADARs ayant des caractéristiques similaires telles que les aéroports ou les ports provoquant une superposition de leurs impulsions.

La deuxième approche, nommée IHACOT, intègre une étape intermédiaire dans l'algorithme HACOT pour traiter des signaux plus complexes. La méthode est tout d'abord basée sur un clustering tridimensionnel à partir du temps d'arrivée, de la fréquence et de la durée d'impulsion afin de mieux séparer les impulsions. Une première phase de fusion des clusters est appliquée avec un clustering agglomératif hiérarchique à partir de la fréquence. Enfin, le clustering agglomératif hiérarchique utilisant des distances de transport optimales mentionné ci-dessus est appliqué pour regrouper les derniers clusters.

Les résultats sont représentés hiérarchiquement et deux modèles de décision ont été développé pour déterminer où arrêter les agrégations. Le premier repose principalement sur des métriques statistiques non supervisées (score de silhouette, score de Calinski-Harabsz et score de Davies Bouldin) ainsi que sur l'analyse des distances de transport optimales afin de fournir un seuil d'élagage unique. Le second modèle est basé sur l'utilisation du test de Kolmogorov Smirnov. Ce modèle applique le test à chaque itération de l'algorithme hiérarchique entre les clusters aggrégés pour déterminer si les impulsions des ces deux clusters appartiennent au même RADAR. Ce processus permet d'obtenir d'obtenir un élagage personnalisée. Des valeurs standard telles que 1% ou 5% sont couramment utilisées pour comparer la valeur du test et déterminer sa significativité. Les signaux RADAR diffèrent, ce qui rend impossible de fixer un seuil unique et standard pour tous les signaux ; une nouvelle méthodologie plus efficace est introduite pour définir un seuil personnalisé pour chaque signal en triant les valeurs des tests obtenues à chaque itération des algorithmes hiérarchiques pour déterminer un point d'arrêt permettant de définir le seuil d'évaluation du test.

Une fois la phase de désentrelacement terminée, lorsque les impulsions ont été correctement séparées, la deuxième étape du processus de reconnaissance RADAR procède à l'identification des émetteurs. La méthode d'identification, nommée IDOT, est basée sur une nouvelle méthodologie supervisée pour identifier les émetteurs dans un signal intercepté avec les distances de transport optimales à partir d'une base de données de référence contenant les caractéristiques d'une multitude de RADAR, notamment en modélisant les pertes d'impulsions du signal. L'algorithme d'identification est basé sur l'élaboration d'une distance entre un ensemble d'impulsions et de classes d'émetteurs via un transport optimal pour identifier les émetteurs à partir de la fréquence, de la durée d'impulsion et une modélisation innovante de la période de répétition des impulsions (PRI) le rendant insensible aux impulsions manquantes.

Lorsqu'un signal regroupe les impulsions mélangées de différents émetteurs, il n'est pas garanti que deux impulsions successives appartiennent au même RADAR (mauvaise estimation du signal, impulsions mal désentrelacées ou identifiées comme aberrantes), rendant impossible l'obtention du bon motif de PRI du RADAR à partir du signal. On obtient donc une distribution différente : la DTOA (également appelé PRI estimée). Pour avoir du sens, la PRI doit être calculé à partir des impulsions appartenant au même émetteur. La PRI a donc été modélisé pour prendre en compte ce phénomène et rendre la méthode d'identification robuste aux impulsions manquantes. Avant de procéder à l'identification, les distances de transport optimales ont été utilisées pour analyser la similarité entre les classes d'émetteurs, fournissant un indicateur puissant du niveau de confiance dans les résultats obtenus aux opérateurs ESM.

Après avoir présenté un processus de reconnaissance RADAR complet, plusieurs expérimentations ont été faites à partir d'un signal simulé en manipulant plusieurs paramètres (valeurs aberrantes, bruit, impulsions manquantes, mélange d'impulsions) pour d'évaluer la robustesse des méthodes développées. Les performances des algorithmes sont mesurés à l'aide de plusieurs métriques. Généralement, l'identification est réalisée simultanément durant l'étape de désentrelacement, mais dans ces recherches, les deux phases sont présentées comme des processus indépendants.

Des comparaisons ont été faites avec d'autres algorithmes de la littérature. Les résultats ont montré que les méthodes de désentrelacement fournissent des résultats encourageants malgré la dégradation de tous ces paramètres, tout comme la modélisation du PRI. Cependant, des conclusions mitigées sont faites lors de l'analyse des résultats de l'algorithme IDOT au cours de l'expérience du mélange d'impulsions et de l'ajout de valeurs aberrantes. Ces observations peuvent être expliquées par le fait qu'une base de données regroupant des caractéristiques très similaires d'émetteurs a été utilisée.

Ces travaux ont constitués une base pour étendre les méthodologies développées; plusieurs axes d'amélioration sont à l'étude. Comme expliqué précédemment, les approches de désentrelacement reposent sur des caractéristiques accessibles et fiables. Les résultats du désentrelacement pourraient être améliorés en incluant des caractéristiques supplémentaires, comme la direction d'arrivée, qui n'est pas toujours accessible mais très discriminante. Un algorithme complémentaire pourrait être développé sur la base de caractéristiques temporairement disponibles et combiné à la méthodologie principale pour améliorer le désentrelacement d'un signal. Enfin, l'application de l'algorithme de clustering hiérarchique utilisant les distances de transport optimales suppose que les clusters appartenant à un émetteur soient simultanément actifs, ce qui dans certains cas n'est pas valide.

Lors de la définition du seuil de confiance pour arrêter les agrégations du dendrogramme durant l'approche hiérarchique, les valeurs du test ont été triées pour identifier la première rupture pour définir ce seuil. Cependant, des cassures plus significatives apparaissent plus tard; il pourrait être intéressant d'analyser l'origine de ces cassures.

Les méthodes de désentrelacement reposent sur l'utilisation de l'algorithme HDBSCAN pour initialiser la séparation des impulsions. HDBSCAN peut identifier les valeurs aberrantes et ainsi exclure des impulsions spécifiques de l'analyse. Cependant, les impulsions provenant d'émetteurs ayant de très faibles émissions au fil du temps peuvent être classées dans ce groupe. Une approche méthodologique pourrait être développée pour gérer ces valeurs aberrantes et vérifier si elles correspondent à des impulsions provenant d'un autre émetteur.

Dans le contexte de la guerre électronique, les opérateurs ESM sélectionnent délibérément les zones d'écoute ; des zones spécifiques comme les aéroports ou les ports ne sont généralement pas privilégiées pour ces activités, bien que ces emplacements puissent potentiellement capter les impulsions de nombreux émetteurs. Ces recherches n'ont pas pris en compte les scénarios dans lesquels une centaine d'émetteurs ou plus pourraient se trouver dans la zone de surveillance. Néanmoins, une solution potentielle pourrait consister à construire une approche semi-supervisée qui impliquerait d'établir et d'utiliser une base de données de référence dans la zone d'écoute pour distinguer les émetteurs reconnus, laissant les impulsions restantes pour un traitement ultérieur.

Concernant l'identification, l'algorithme développé classe les émetteurs RADAR connus appartenant à une base de données de référence. Détecter les nouveaux émetteurs et enrichir automatiquement cette base de données pour proposer une solution complète et non supervisée d'identification des émetteurs pourrait être une extension passionnante de ce travail. La méthodologie d'identification repose uniquement sur trois caractéristiques fiables et disponibles ; d'autres fonctionnalités décrivant les émetteurs pourraient être ajoutées pour compléter et renforcer l'identification lorsqu'elles sont disponibles.

L'algorithme d'identification représente sous forme de distribution Gaussienne les émetteurs ayant une PRI aléatoire et par un mélange de distributions gaussiennes lorsque l'on considère le taux d'impulsions manquant. Lorsque le taux d'impulsions manquant est élevé, l'algorithme a tendance à confondre l'ensemble de distribution des impulsions avec la distribution de la classe d'émetteur ayant un PRI aléatoire. L'exploration de méthodes alternatives pour représenter les ensembles d'impulsions pourrait être bénéfique, garantissant que les tendances des données sont toujours capturées efficacement.

De façon similaire à la modélisation du PRI en présence d'impulsions manquantes, la distribution de fréquence pourrait être modélisée en fonction de différents niveaux de valeurs aberrantes. Cette approche renforcerait les résultats de l'algorithme d'identification car la fréquence est une caractéristique très discriminante.

Une version basique d'un simulateur de données a été présentée pour évaluer les résultats et valider les méthodes développées. Une réflexion plus approfondie pourrait être menée sur le simulateur afin d'en accroître la complexité et de fournir des signaux RADAR plus réalistes et plus complexes. Le développement d'algorithmes en temps réel pour le désentrelacement des signaux joue un rôle crucial dans la guerre électronique, et ces algorithmes permettent aux opérateurs ESM d'extraire rapidement des informations et des composants électroniques vitaux en utilisant des capacités de prise de décision rapides. Les différentes méthodes développées pourront être intégrées dans des systèmes embarqués pour être testées et évaluées en temps réel.

Dans ces recherches, l'hypothèse selon laquelle aucune mesure adverse introduisant des signaux perturbateurs a été faite. Une analyse plus approfondie pourrait être effectuée pour évaluer la robustesse de ces méthodes face à de telles techniques de perturbation.

Ces recherches proposent de nouvelles méthodologies non supervisées utilisant des distances de transport optimales pour désentrelacer un signal RADAR et identifier les émetteurs présents. Une attention particulière a été portée à la simulation des données, élément important dans la construction des méthodes. Plusieurs axes d'amélioration sont proposés pour poursuivre ces recherches et améliorer les résultats.

La fin d'une méthodologie innovante symbolise le début de l'exploration de solutions encore plus innovantes.

Introduction

Military intelligence is a significant concern for the security and defense of a country. In a constantly changing world, the issues related to national defense and security are complex and diversified, such as the prevention of external threats, terrorism, piracy, and the proliferation of mass destruction weapons, requiring the implementation of preventive methods and appropriate strategies to deal with constantly evolving threats. The information collected enables institutions to swiftly respond, manage, and anticipate national and international crises.

This challenge is notably apparent in electronic intelligence (ELINT) [Wil06; Sko08; Sch80], where the emphasis is on collecting and using electromagnetic signals from various sources to extract sensitive information for guiding operators to commit strategically. RADAR systems are frequently required to detect threats and implement appropriate countermeasures, specifically with passive systems offering discrete listening and data collection approaches at a lower cost.

Detection of RADAR emissions is essential for identifying threats, monitoring enemy movements, protecting friendly forces, and providing a defensive strategy. However, the challenges are numerous, primarily due to the diversity of the signals, the sophisticated modulation techniques, and the evasion measures employed to avoid detection. Recent technological developments have led to many innovations in defense due to the increasing complexity and sophistication of electronic equipment; RADAR emitters have more similar electromagnetic spectra, making their differentiation complex. This challenge is more pronounced as the intercepted signals are voluminous and contain more information to be processed.

Artificial intelligence (AI) allows machines to adapt, develop, and exploit their algorithms swiftly, providing a significant advantage to organizations capable of mastering the tools and transforming the RADAR signal processing chain. All these changes represent a continuous challenge and require advanced technical expertise, a deep understanding of RADAR systems, and constant adaptation to technological developments to provide new, more precise techniques for separating received pulses and identifying RADAR emitters from an intercepted signal.

Among all the machine learning algorithms, optimal transport theory [Vil09; Mon81; GM96] provides simple and powerful tools to solve many optimization problems; optimal transport is a classic problem in applied mathematics that seeks to determine the most efficient way to move quantities from a source to a target, considering specific constraints. This approach finds applications in various fields such as computer vision, adaptation domain, text generation, or distance generation.

This research aims to respond to the increasing need to develop advanced machine learning methods for passive RADAR intelligence, specifically to improve the RADAR Recognition Process with optimal transport distances. The RADAR Recognition Process can be defined in two main stages: the first stage involves deinterleaving a signal by separating and grouping the mixed pulses of an unknown number of emitters before identifying them in the second stage. Deinterleaving a signal is a critical stage in the RADAR processing chain. The Electronic Support Measures (ESM) operators need to extract information on the intercepted pulses regarding the quality of the estimated pulses, number of emitters present, noise rate, etc. The processing is done in an unsupervised setting, resulting in critical decision-making. Based on a received signal containing pulses from an unknown number of emitters, two new, simple, unsupervised approaches based on a combination of clustering algorithms and optimal transport distances are introduced to deinterleave a signal. The first step consists of separating the pulses with a 2-dimensional clustering algorithm under the constraint that the pulses of two different emitters cannot belong to the same cluster. Then, as the emitters exhibit complex behavior and can be represented by several clusters, a hierarchical agglomerative clustering based on an optimal transport distance (HACOT) is proposed to merge these clusters. A variant is also developed, capable of handling more complex signals with similar characteristics. It integrates a pre-processing step with a 3-dimensional clustering to separate pulses before grouping them with a simple hierarchical agglomerative clustering (HAC) based on a Euclidean distance. Finally, the above-mentioned hierarchical agglomerative clustering using optimal transport distances is applied to group the last clusters.

From the deinterleaving outputs and by assuming that the sets of pulses are composed of the pulses from a single emitter, the second stage of the RADAR reconnaissance process is performed by elaborating a distance with optimal transport between a set of pulses and the describing characteristics of an emitter from a reference database. Although electronic warfare (EW) [Poi12; FS08; Hoi80] theaters go hand in hand with countermeasures, this work is positioned in an ELINT framework and concerns only the waves emitted by emitters to characterize them. The presence of adversary measures introducing disruptive signals is not considered.

This manuscript mainly introduces a new RADAR Recognition Process based on optimal transport distances. It comprises five main chapters introducing and developing the proposed approaches and proposing a simulated data environment. Chapter 1 provides the basic knowledge for understanding the content, references the main algorithms used to solve the problem, and describes their functioning. Other relevant algorithms used throughout this manuscript are also presented to facilitate understanding the choices made. Illustrations will be used to comprehend their utility in our problem.

As previously explained, RADAR data is a fundamental aspect of this thesis. Hence, Chapter 2 emphasizes the RADAR system environment used and developed to build our methodology, from the essential operation of an emitter through the simulation of pulses to their modeling. A data simulator was provided in collaboration with Avantix. Due to its confidentiality, a public simulator is proposed and described, which includes more comprehensive details. Examples of emitters illustrate the simulator's functioning and generate a simulated RADAR signal with specific modelization of emitter characteristics. Finally, an overview of the existing methods to process RADAR pulses is presented, and their main limitations are analyzed to understand why they are unsuitable for the deinterleaving and identification problem.

From the previously developed simulated environment, Chapter 3 introduces the first step of the RADAR Recognition Process with two unsupervised methodologies based on clustering algorithms and optimal transport distances to deinterleave a RADAR signal and two efficient approaches to prune the dendrogram during the hierarchical approach. Pruning a dendrogram represented a significant challenge; stopping aggregations is specific to the dataset and the objectives; two approaches have been developed to account for this problem. The first is based on unsupervised metrics and knowledge of the RADAR system environment. Then, a second approach uses powerful and simple statistical tests to compare cluster distributions, group them, and get personalized pruning.

After deinterleaving the signal, Chapter 4 presents a novel method to identify the emitter in a signal based on optimal transport distances. The algorithm is based on innovative pulse repetition interval (PRI) modeling and is insensitive to missing pulses, allowing the use of pulse repetition of interval characteristics.

Chapter 5 introduces simulation results highlighting the developed approaches' performances and robustness. The two steps of the RADAR recognition process are initially evaluated separately: existing methods and a simulated signal under varying conditions challenging the deinterleaving methods by manipulating the outliers, noise, or missing pulse rates. Several simulated signals are used to illustrate the experiments. Similarly, the identification method is evaluated by deteriorating the signal quality and analyzing the identification results. Afterward, the entire process is evaluated by applying the different algorithms developed with several metrics. Finally, proximity analyses are presented between the emitters to provide a complete analysis.

Finally, the concluding Chapter reminds the essential contributions of this research work and the potential prospects for improving the RADAR Recognition Process. A list of publications is given at the end of the manuscript.

1 - Mathematical formulation of the algorithms

This chapter provides the essential knowledge for understanding the content, references the main algorithms used to construct a RADAR Recognition Process, and describes their functioning. Other relevant algorithms cited throughout this manuscript are also presented. The first section presents the basics and provides the mathematical tools of optimal transport theory. Although the approaches are mainly based on the Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) algorithm and Hierarchical Agglomerative Clustering (HAC), the clustering algorithms section will present other conventional clustering algorithms using examples to illustrate them. The methods developed mainly provide results in the form of dendrograms. Based on unsupervised measurements and statistical tests, several decision models were created to determine where to stop aggregations in these dendrograms. Several sections are dedicated to the presentation of all these tools before presenting supervised metrics applied to evaluate the quality of the proposed approaches.

1.1 . Optimal Transport

This manuscript presents two methods to deinterleave a RADAR signal in Chapter 3, an algorithm identifying the emitters in Chapter 4, and a proximity analysis between the emitters in Section 4.4. These methods are mainly based on comparing multiple distributions to measure their similarity or dissimilarity. The data used to develop the approaches is represented in various forms, such as histograms, probability distributions, or mass densities, requiring a metric to compare the geometry of several distributions. This Section demonstrates that optimal transport (OT) distances [Vil09; Fla+21; GM96] are particularly well-suited.

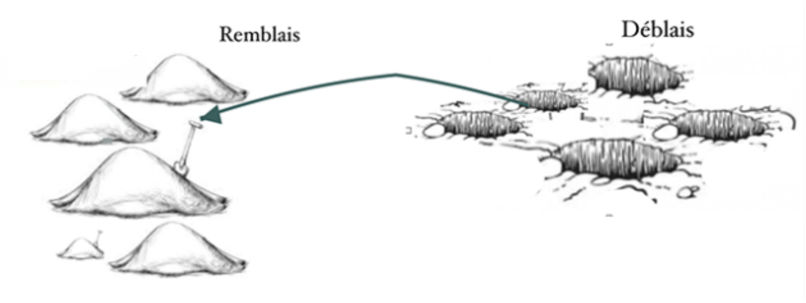
Several ways exist to define the distance between probability distributions. For instance, the Kullback-Leibler divergence or the total variation distance is frequently used as other measurements shown in Table 1.1.

	Discrete case	**Continuous case**
Kullback-Leibler Divergence	$\sum_{k=1}^{K} P(k)\log\left(\frac{P(k)}{Q(k)}\right)$	$\int_{\mathbb{R}} p(x)\log\left(\frac{p(x)}{q(x)}\right)dx$
Total variation distance	$\frac{1}{2}\sum_{k=1}^{K} \lvert P(k)-Q(k)\rvert$	$\frac{1}{2}\int_{\mathbb{R}} \lvert p(x)-q(x)\rvert dx$
Hellinger Distance	$\sqrt{\frac{1}{2}\sum_{k=1}^{K}\left(\sqrt{P(k)}-\sqrt{Q(k)}\right)^2}$	$\sqrt{\frac{1}{2}\int_{\mathbb{R}}\left(\sqrt{p(x)}-\sqrt{q(x)}\right)^2 dx}$
Bhattacharyya Distance	$-\log\left(\sum_{k=1}^{K}\sqrt{P(k)Q(k)}\right)$	$-\log\left(\int_{\mathbb{R}}\sqrt{p(x)q(x)}dx\right)$

Table 1.1: Measurements between 2 probability distributions P and Q.

These distances or divergences cannot be used when the distributions representing the data have different dimensions, but above all, they can not compare probability laws with disjoint supports. By applying their definitions, calculating the similarity or dissimilarity between the probability distributions will return by constructing a large value. Conversely, optimal transport finds the optimal correspondence between the elements of the two distributions and minimizes the total cost of moving the mass. The optimal transport distances [CPD20; GM96; Cut13] are used to overcome these limitations as they provide a powerful way to deal with multiple and different distributions.

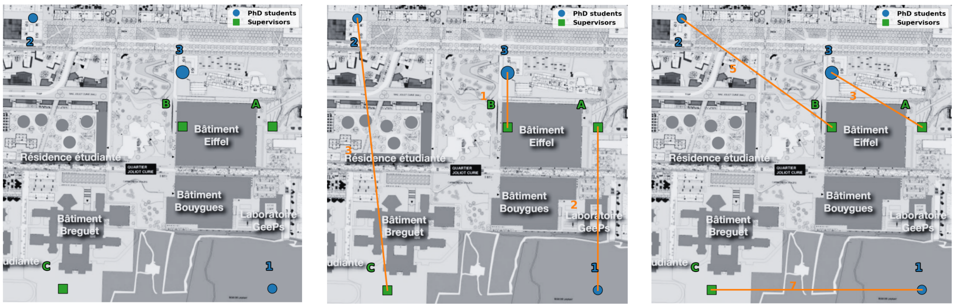
The concept of optimal transport has been studied for many years, even dating back to the 18th century, with one of its first formulations attributed to Gaspard Monge and reformulated by Kantorovich in 1942. Gaspard Monge (1746-1818), a French mathematician and engineer, is well-known for pioneering descriptive geometry, revolutionizing the graphic representation of three-dimensional objects.



His work significantly influenced the fields of mathematics and engineering, and he also contributed to the reform of science education in France. Monge is mainly recognized for his contributions in his memories [Mon81] by laying the foundations for optimal transport theory. Although Monge did not explicitly introduce the term "optimal transport," his ideas on how to efficiently move resources from one place to another contributed to the emergence of this concept, which aims to find the most economical way to move given masses from one location to another while minimizing a cost. Applications of significant note can be found in fields such as economics, logistics, and data science.

Monge studied the optimal transport to move ground quantity from one place to another as a bijection of the points. He assumed that the product of mass and distance gave the cost of transporting a unit of mass over a distance. Monge postulates that all displacements have different costs, and for all these possible transports, there is one called optimal transport, which minimizes the total cost.

Let us consider the simple example of three students studying at CentraleSupélec seeking to have their articles corrected by supervisors. The students have each written an article they wish to publish in a reputable journal. To optimize their chances of publication, students must have their articles reviewed and corrected by supervisors, seeking their opinions before submission. The students have their offices in different locations on campus, as do their supervisors, as shown in Figure 1.1a. One can calculate the distance between the students' and supervisors' offices by employing, for instance, Euclidean distance.



(a) Locations of the offices. (b) Total cost = 2 (c) Total cost = 5

Figure 1.1: Examples of displacements between Ph.D. students and their supervisors' offices at the CentraleSupélec campus. The orange lines represent the displacements.

There are several ways for students to see their supervisors, as illustrated in Figures 1.1b and 1.1c. Each displacement does not require the same effort for students as their offices are more or less far from their supervisors', as shown in Figures 1.1b and 1.1c. Knowing the distances between offices, the total cost of the students' displacements can be calculated as the sum of the displacements of all students from their offices towards that of their supervisors. Among all these displacements, there is one that is minimal and optimizes students' displacements. In this example, this movement is called optimal transport and is symbolized in Figure 1.1b.

Leonid Kantorovich, a 20th-century economist and mathematician, made notable contributions to optimal transport theory, developing a mathematical reformulation [Rac85] of Monge's formulation. Kantorovich expanded on Monge's initial research, developing a different approach to solving transportation problems by introducing more formal mathematical aspects. Kantorovich introduced a probabilistic version of optimal transport considering differences between the quantities of departure and arrival, *e.g.*, splitting the masses. It considers plan of the couplings between the different points and solves a convex linear optimization problem. Kantorovich's work greatly expanded and generalized Monge's view of optimal transport theory by introducing linear programming and proposing a method to minimize transportation costs while considering various constraints.

Consider students' examples once more. In this example shown in Figure 1.2a, four students seek correction for their articles, and only two supervisors are involved. However, these two supervisors have different availability times, and not all Students need to spend the same time with a supervisor. Supervisor A has only 30 minutes, while Supervisor B has 3 hours; Students 1 and 2 have completed writing their articles, meaning they do not need much time, while Student 4 has just started writing and needs more help. According to Monge's formulation, finding a transport plan is not possible.

From a mathematical perspective, the students' and the supervisors' offices can be represented as discrete probability measures:

$$\mu_s = \sum_{n=1}^{N_s} \alpha_n \delta_{x_n} \text{ and } \mu_t = \sum_{m=1}^{N_t} \beta_m \delta_{y_m}, \tag{1.1}$$

with $\alpha = (\alpha_1, \dots \alpha_{N_s})^T \in \mathbb{R}_+^{N_s}$, and $\beta = (\beta_1, \dots \beta_{N_t})^T \in \mathbb{R}_+^{N_t}$ such as:

$$\sum_{n=1}^{N_s} \alpha_n = \sum_{m=1}^{N_t} \beta_m = 1. \tag{1.2}$$

A transport plan $\mathbf{P}$ between μ_s and μ_t is defined by its coefficients $\mathbf{P}_{nm}$, representing the amount of mass taken from x_n to y_m with $c(\cdot,\cdot)$ a cost function, as $c(x,y) = \|x - y\|_2$ for example, and $C_{nm} = c(x_n, y_m)$ the cost of transporting a unit of mass from x_n to y_m. The total cost $C(\mathbf{P})$ of a transport plan $\mathbf{P}$ is:

$$\begin{aligned} C(\mathbf{P}) &= \sum_{n=1}^{N_s} \sum_{m=1}^{N_t} c(x_n, y_m)\mathbf{P}(x_n, y_m) \\ &= \langle \mathbf{C}, \mathbf{P} \rangle . \end{aligned} \tag{1.3}$$

Then, the optimal transport plan $\mathbf{P}^\star$ is obtained as the minimizer of:

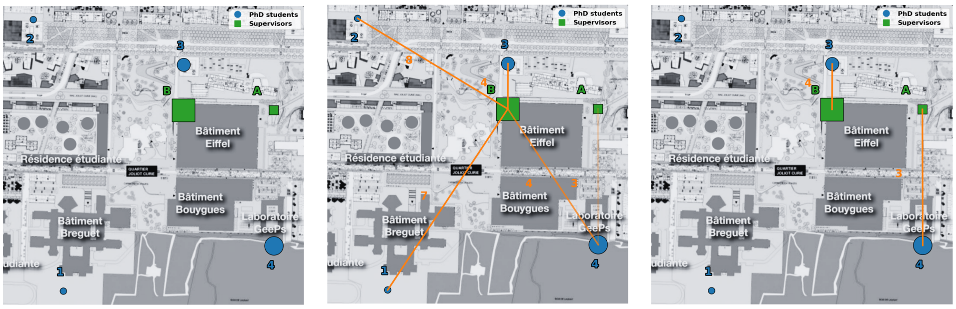
$$\mathbf{P}^\star = \underset{\mathbf{P} \in \Pi(\mu_s, \mu_t)}{\operatorname{argmin}} \ \langle \mathbf{C}, \mathbf{P} \rangle . \tag{1.4}$$

Under the following constraints to guarantee the consistency of the transport plan $\mathbf{P}$ between μ_s and μ_t:

$$\Pi(\mu_s, \mu_t) = \left\{ \mathbf{P} \in (\mathbb{R}^+)^{N_s \times N_t}, \mathbf{P}\mathbf{1}_{N_t} = \alpha \text{ and } \mathbf{P}^T\mathbf{1}_{N_s} = \beta \right\}. \tag{1.5}$$

The optimal transport distance between μ_s and μ_t is then defined by $d(\mu_s, \mu_t) = C(\mathbf{P}^\star)$.

The only results presented here concern discrete cases since this manuscript mainly uses discrete functions. The results for the continuous case are obtained similarly to those for the discrete case.



(a) Locations of the offices. (b) Kantorovich relaxation with a cost of 4.35. (c) Unbalanced optimal transport with a cost of 1.45.

Figure 1.2: Examples of displacements between Ph.D. students and their supervisors' offices at the CentraleSupélec campus when supervisors do not have the same availability. The square size indicates the availability of supervisors, and the orange lines represent the displacements under constraints.

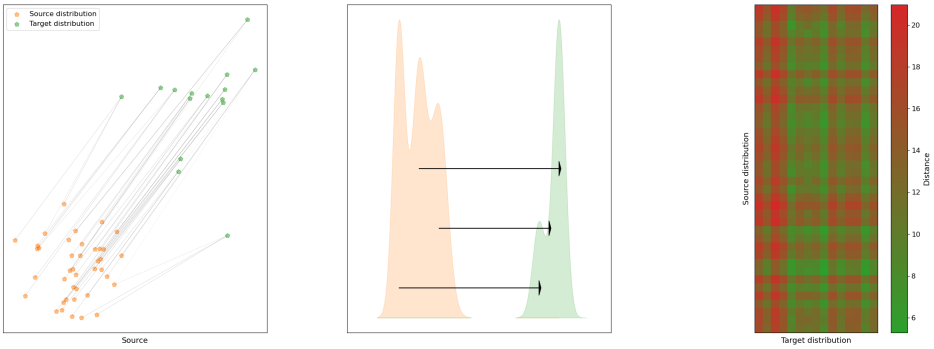
According to Kantorovich's reformulation, it is possible to find transport plans from Figure 1.2a but also an optimal one as illustrated in Figure 1.2b. In this example, with limited availability, Supervisor A will exclusively help Student 4, while Supervisor B will help all the students because he has more time available.

Note that if the cost function between two points is the distance between them, i.e., $c(x, y) = \|x - y\|^p$, the optimal cost is equivalent to the first-order Wasserstein distance W_1, also called the Earth Mover's Distance [RTG00; WB19].

By adding $c(x, y) = \|x - y\|^p$ in Equation (1.4), one have:

$$W_p^p(\mu_s, \mu_t) = \sum_{n=1}^{N_s} \sum_{m=1}^{N_t} \|x_n - y_m\|^p \mathbf{P}^\star(x_n, y_m). \tag{1.6}$$

This problem has been extended and applied to many fields, including mathematics, by defining a distance for shifting distributions as shown in Figure 1.3b. The optimal transport interpolates the two distributions by capturing the geometric phenomenon underlying the spaces and considering the proximity between the masses. The distribution source and target are displayed with the movements made by each point on the left plot. The source distribution points are sent in different directions considering the proportions of the target distribution points, explaining that some points are sent in the same localization. The cost matrix is obtained by calculating the transport cost between each point. The y-axis represents the points from the source distribution, while the x-axis is the points from the target distribution. The colorimetry of the cost matrix on the right plot shows the displacements made by the points; the red color signifies a high transport cost between the two points, indicating that the point from the source distribution is close to the target distribution's point. Conversely, the green indicates a high transport cost, implying the two considering points are distant.



(a) Optimal Transport plan. (b) Mass displacements. (c) Cost matrix.

Figure 1.3: Transport plan between two samples and the associated cost matrix between each point. Red indicates a high transport cost, while green indicates a lost transport cost.

Figure 1.4 displays an example of applying optimal transport to define a distance between distributions with identical or disjoint supports. The displacement between the source (in the middle) and two targets is analyzed. Source and Target 1 have similar appearances, resulting in very few moves to coincide Source points toward Target 1 points, producing a lower transport cost. Conversely, Source points require significant movement to coincide with Target 2, implying a high transport cost. Optimal transport is utilized to establish a distance; the distance between the Source and Target 1 distribution is set as low, while it will be high between the Source and Target 2 distribution.

The optimal transport formulations imply that all masses must be transported from μ_s to μ_t. As a result, optimal transport is not robust to noise, outliers, and local mass variations [Fat+21]. An extension of the optimal transport version, named optimal unbalanced transport (UOT) [Ben03; Chi+15; LMS18], has been developed to overcome these limitations. This version can cope with unbalanced measures by allowing the destruction or the creation of masses (i.e., allowing the variation of the total mass). For that, a relaxation of the

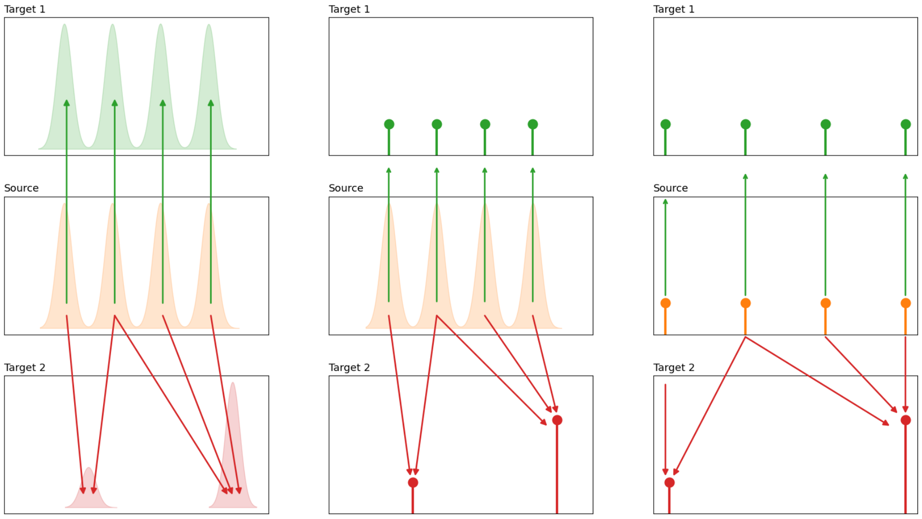


Figure 1.4: Optimal transport process between distributions.

marginal constraints has been proposed.

For the discrete case writing in Equation (1.4), the marginal constraints are replaced by convex functions ψ_1 and ψ_1:

$$\mathbf{P}^\star = \operatorname*{argmin}_{\mathbf{P}\in\mathbb{R}^{\Omega_s\times\Omega_t}} \sum_{n=1}^{N_s}\sum_{m=1}^{N_t} c(x_n, y_m)\mathbf{P}(x_n, y_m) + \psi_1\left(\sum_{n=1}^{N_s}\mathbf{P}(x_n, y_m)\right) + \psi_2\left(\sum_{m=1}^{N_t}\mathbf{P}(x_n, y_m)\right). \quad (1.7)$$

Figure 1.2c illustrates this variant; as Students 1 and 2 have almost completed their project, it is more optimal for them not to consult the supervisors and let them devote their time to the other students. The transport cost is three times lower.

For some applications, using the OT can be problematic and requires making specific corrections to find a good compromise between the computational speed and the quality of results, specifically in large-scale OT problems. For example, when dealing with high-dimensional spaces, the curse of dimensionality strongly impacts the OT (sparse data, computational complexity, data dilution, etc.). A fast algorithm is required to process data in real-time. In many applications, processing data in real-time with maximum efficiency is imperative to make decisions quickly. For example, data must be rapidly processed in military surveillance and reconnaissance operations to provide real-time intelligence on enemy movements and potential threats.

For this, a corrected version of OT was proposed, named the Entropy-regularized Optimal Transport (EOT), which integrates a regularization function to speed up the optimal transport [Cut13; LMS18; DPR18]. Let us consider with $\rho > 0$ a regularized parameter, $H(\mathbf{P})$ an entropic function.

By using Equation (1.4), one have:

$$\mathbf{P}^{\star} = \underset{\mathbf{P}\in\Pi(\mu_s,\mu_t)}{\operatorname{argmin}} \sum_{n=1}^{N_s}\sum_{m=1}^{N_t} c(x_n, y_m)\mathbf{P}(x_n, y_m) + \rho H(\mathbf{P}), \tag{1.8}$$

with $H(\mathbf{P})$ an entropic-regularization function for example such as:

$$H(\mathbf{P}) = \sum_{n=1}^{N_s}\sum_{m=1}^{N_t} \mathbf{P}(x_n, y_m)\log(\mathbf{P}(x_n, y_m)). \tag{1.9}$$

There are many other applications of optimal transport, such as image processing [ACB17; Cou+17; SS15; RP15], Natural Langage Processing [AJ18], objects matching [Mém11]. Moreover, many extensions of optimal transport have been proposed in recent years, including the Gromov-Wasserstein distance, allowing two metric spaces to be compared [PCS16] or new algorithm classes for applying OT on geometric domains [Sol+15].

1.2 . Clustering

This section references the main partitioning algorithms used to solve the problem and describes their functioning. Other relevant partitioning algorithms are also presented. Clustering provides powerful unsupervised tools to discover and learn data structure and observation similarity patterns without any by extracting meaningful or useful classes. Table 1.2 summarizes the differences between the algorithms presented.

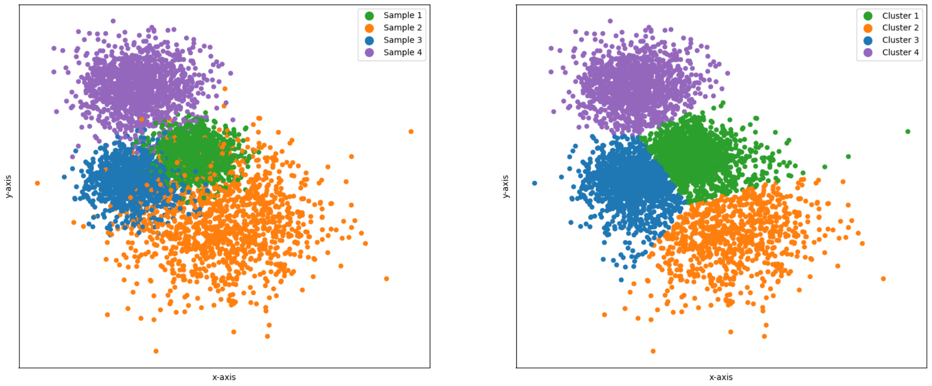
1.2.1 . K-Means

The K-means algorithm [HW79; Mac+67] is one of the most popular clustering algorithms. K-means, illustrated in Algorithm 1, randomly identifies the k centers from the data and assigns the closest points to each center by minimizing the sum of the distances between each point and the K centroids. The process is iterated until convergence.

K-means provides quick and easy-to-interpret results and handles spherical data but is limited in finding irregular data structures. The number of clusters must be specified to run the algorithm. Several methods, such as the silhouette score or the elbow method, could determine this optimal number. When the dataset is simple, these methods give correct results but are limited to unstructured data, leading to the fixing of an erroneous number. The algorithm cannot detect outliers and classifies all points, making it very sensitive to noise. Figure 1.5b represents the results of KMEANS applied on Figure 1.5a. Data from sample 2 (orange) is mixed with data from Samples 1 and 2 (blue and green). This mixture can be problematic in application cases such as the defense sector. Mixing data can lead, for example, in the case of classification, to a poor representation of the data distribution and, therefore, unsatisfactory identification.

Algorithm 1 K-MEANS Algorithm.

```
Input :
Number of clusters to identify, $K$
Set of points, $\mathcal{X} = \{x_1, x_2, x_3..., x_N\}$
Initialization : Random selection of $K$ points as centroids from $\mathcal{X}$ :
    $\mathcal{C} = \{c_1, c_2, c_3..., c_K\}$
Procedure :
    • Assign each point to its closest centroids: $k = \text{argmin}_k \|c_k - x_i\|^2$ , $\forall i \in \{1, ..., N\}$
    • Update cluster centroids: $c_k = \frac{1}{n_k} \sum^{i \in c_k} x_i$
      with $n_k = \#c_k$
    • Iterate the previous steps until convergence
Output : $\mathcal{Y} = \{y_1, ..., y_N\}$: points labels of $\mathcal{X}$
```



(a) Simulated observations with colors identifying true labels.

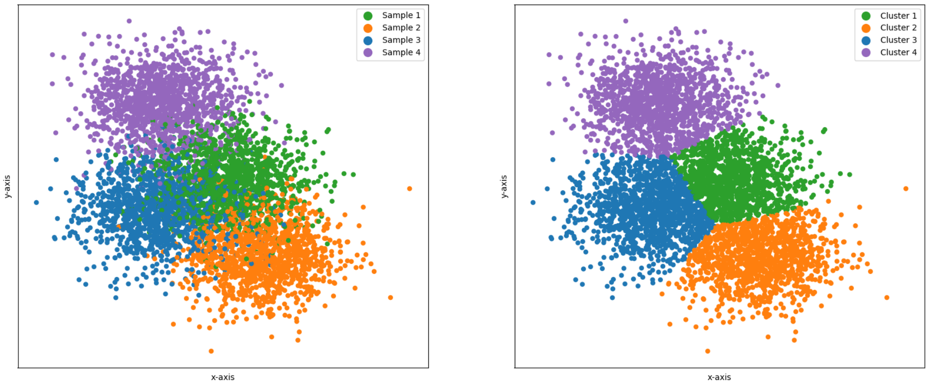
(b) Kmeans outputs with colors identifying clusters.

Figure 1.5: K-means application on a simulated dataset.

1.2.2 . Gaussian Mixtures Models

Gaussian mixture models [MB88] (GMM) are statistical models widely used to detect patterns in data. The clustering methodology, highlighted in Algorithm 2, is similar to the K-means algorithm except that Gaussian density functions characterize the detected clusters; the model assigns to each point a probability to belong to each Gaussian distribution. The algorithm estimates the dataset's average, covariances, and weights of K Gaussian distribution. The Maximum Likelihood Estimation (MLE) [DLR77; MK07] approach estimates the model parameters. Firstly, each point is assigned to a cluster according to the likelihood of each Gaussian. Then, the Gaussian distribution parameters are updated until convergence, and the model assigns each point to a cluster.

GMMs allow the modeling of various distributions, can model clusters of different sizes, and are robust to noise. However, GMMs require the number of clusters to be specified beforehand, such as the K-means Algorithm, which could be complicated to fix and have a long execution time, specifically when searching for many clusters. GFigure 1.6b represents the results of GMM applied on Figure 1.6a. Samples 1 and 3 (blue and green) are mixed into Clusters 1 and 3 (blue and green). All points have been classified. GMMs cannot detect outliers. This can be problematic when representing data as a probability distribution because the distribution may not be representative.



(a) Simulated observations with colors identifying true labels.

(b) GMM outputs with colors identifying clusters.

Figure 1.6: GMM application on a simulated dataset.

Algorithm 2 MLE Approach for Gaussian Mixtures Models.

Input :
Number of mixture component, K
Set of points, $\mathcal{X} = \{x_1, x_2, ..., x_N\}$
Initialization : Randomly initialisation of γ_k, σ_k^2 and π_k
Procedure :

- **E-step**: compute responsibilities

$$\widehat{\gamma_{nk}} = \frac{\widehat{\pi}_k p(x_n|\theta_k)}{\sum_{j=1}^{K} \widehat{\pi}_j p(x_n|\theta_j)}$$

- **M-step**: compute parameters

$$\widehat{\mu}_k = \frac{\sum_{n=1}^{N} \widehat{\gamma_{nk}} x_n}{\widehat{\gamma}_k} \text{ , } \widehat{\pi}_k = \frac{\widehat{\gamma}_k}{N}$$

$$\widehat{\sigma^2}_k = \frac{\sum_{n=1}^{N} \widehat{\gamma_{nk}} (x_n - \widehat{\mu}_k)(x_n - \widehat{\mu}_k)^T}{\widehat{\gamma}_k}$$

- Repeat until convergence

with $p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \sigma_k^2)$ and $\theta = (\mu, \sigma^2)$.

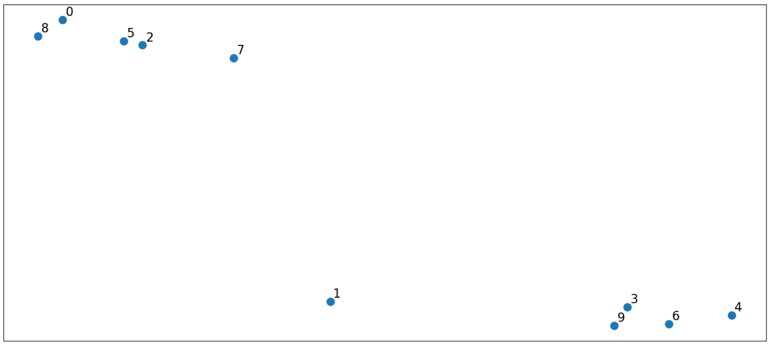
Output : $\mathcal{Y} = \{y_1, ..., y_N\}$: points labels of $\mathcal{X}$

1.2.3 . Hierarchical Agglomerative Clustering.

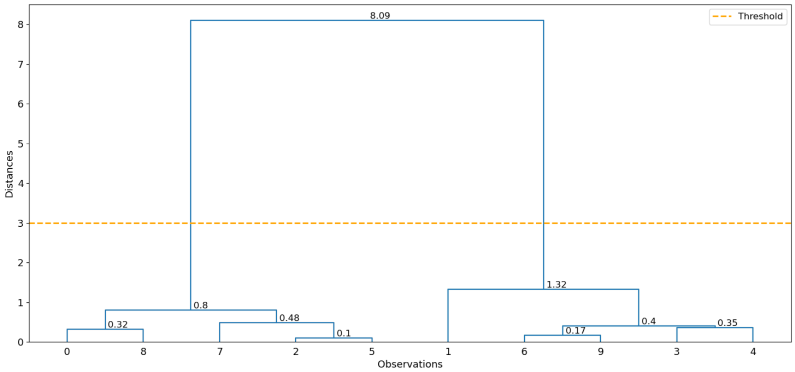
Hierarchical agglomerative clustering Algorithm (HAC) [Joh67] is a clustering method aggregating observations into clusters and organizing them as a set of nested clusters as a hierarchical tree. An example is provided in Figures 1.7 with simulated with 10 observations belonging to 2 different samples identified by a color. The process starts by considering all points as a singleton, computes the distance between clusters, aggregates the two closest clusters, and repeats until a single cluster is obtained. The results are visualized as a dendrogram in Figure 1.7b, and the values displayed are the inter-cluster distances. For example, the metrics presented in Section 1.3 could be used to determine where to stop aggregations. The orange line identifies aggregated groups. The final representation is presented in Figure 1.7c.

Algorithm 3 Hierarchical Agglomerative Clustering Algorithm.

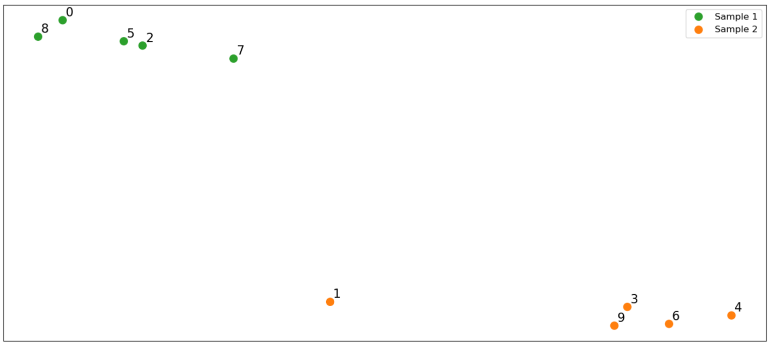
```
Input :
Set of points, $\mathcal{X} = \{x_1, x_2, ..., x_N\}$ with $\#\mathcal{X} = N$
Distance, $dist$

Initialization :
$\mathcal{C}$ with $N$ clusters as singleton from $\mathcal{X}$: $\mathcal{C} = \{c_1, ..., c_N\}$
Compute $\mathcal{D}$, the pairwise distance matrix between all clusters in $\mathcal{C}$

Procedure :
While $\#\mathcal{C} > 1$:
    1. Find two most similar clusters in $\mathcal{D}$: $c_i$ and $c_j$ according to $\operatorname{argmin}_{i,j} dist_{i,j}$
    2. Merge $c_i$ and $c_j$ into a new cluster
    3. Remove $c_i$ and $c_j$ from $\mathcal{C}$ and $dist_{i,j}$ from $\mathcal{D}$
    4. Update the distance between the new cluster and the other clusters in $\mathcal{D}$

Output : $\mathcal{Z}$: Hierarchical structure identifying the merged clusters set
         with the inter-cluster distances.
```



(a) Simulated observations.



(b) Hierarchical Agglomerative Clustering with the Euclidean distance display.



(c) Aggregation results.

Figure 1.7: Illustration of Hierarchical Agglomerative Clustering.

The pairwise methods between points can be computed using multiple methods as:

- Single linkage: uses the minimum distance between two clusters by calculating distances between the two closest points between the merged clusters,

$$d(X, Y) = \min_{x \in X, y \in Y} dist(x, y). \tag{1.10}$$

- Complete linkage: uses the maximum distance between two clusters by calculating distances between the two furthest points between the merged clusters,

$$d(X, Y) = \max_{x \in X, y \in Y} dist(x, y). \tag{1.11}$$

- Average linkage: uses the average distance between all points of the merged clusters,

$$d(X, Y) = \frac{1}{I * J} \sum_{x \in X}^{I} \sum_{y \in Y}^{J} dist(x, y). \tag{1.12}$$

- Centroid linkage: uses the distance between the centroid of the merged clusters,

$$d(X, Y) = dist(C_X, C_Y). \tag{1.13}$$

  With C_X and C_Y, the centroids of the X and Y merged clusters defined as $C_X = \frac{1}{I} \sum_{x \in X}^{I} x$ and $C_X = \frac{1}{J} \sum_{y \in Y}^{J} y$.

Hierarchical algorithms have the advantage of not making assumptions about the underlying structure of the data. They are effective for small datasets, but their performance strongly depends on the choice of pairwise methods and the distance used. Moreover, the main challenge is determining the appropriate cutoff for having the best partition. Finally, these algorithms can be computationally intensive, constituting a limitation in contexts requiring rapid execution.

1.2.4 . Density-Based Spatial Clustering of Applications with Noise

Density-Based Spatial Clustering of Applications with Noise algorithm [Est+96] (DBSCAN) is an unsupervised clustering algorithm based on density search able to detect outliers. The main idea is to regroup points that live in the same dense area. More precisely, the algorithm defines an ϵ-neighbourhood for each point x_i to be clustered, as follows $\mathcal{N}_\epsilon(x_i) = \{x_k \in \mathcal{X} | d(\mathbf{x}_k, \mathbf{x}_i) \leq \epsilon\}$, for a given distance $d(.,.)$. Then, depending on the size of $\mathcal{N}_\epsilon(x_i)$, namely $\#\mathcal{N}_\epsilon(x_i) \geq MinPts$, x_i is either clustered or labelled as an outlier. DBSCAN, illustrated in Algorithm 4, makes no assumptions about the data structure and does not require the cluster number to detect; it can handle irregular data structures. Its ability to detect outliers makes it robust to noise and outliers, thus improving the quality of clustering results. Therefore, DBSCAN is very sensitive finally to the ϵ and $MinPts$ parametrization and the distance used. Note that it is challenging to find ϵ as shown in Figure 1.8.

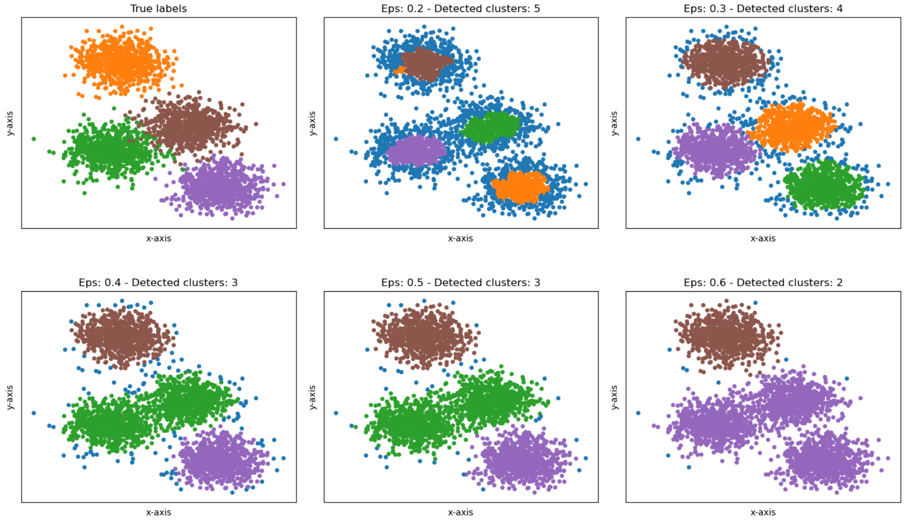


Figure 1.8: HDBSCAN application on a simulated dataset by varying ϵ. Each color represents a cluster, and the blue dots are the outliers.

Algorithm 4 Density-Based Spatial Clustering of Applications with Noise Algorithm.

```
Input :
Set of points to be clustered, $\mathcal{X}$
Minimum number of points to form a cluster, $MinPts$
Radius $\epsilon$

Procedure :
for all $x_i$ of $\mathcal{X}$ do
    Selection of unvisited point by the algorithm $x_i$
    Identify $\mathcal{N}_\epsilon(x_i)$, an $\epsilon$-neighbourhood of $x_i$
    if $\#\mathcal{N}_\epsilon(x_i) \leq MinPts$ then
        Mark $x_i$ as noise
    else
        Create a class $C$ containing $x_i$
        Mark $x_i$ as visited
        for all $x_j$ in $\mathcal{N}_\epsilon(x_i)$ do
            Verify that $x_j$ has not been visited by the algorithm
            Identify $\mathcal{N}_\epsilon(x_j)$, an $\epsilon$-neighbourhood of $x_j$
            if $\#\mathcal{N}_\epsilon(x_j) \geq MinPts$ then
                $\mathcal{N}_\epsilon(x_i) = \mathcal{N}_\epsilon(x_i) \cup \mathcal{N}_\epsilon(x_j)$
            else

            end
            if $x_j$ is not already classified, add $x_j$ to $C$
        end
    end
end
Output : $\mathcal{Y} = \{y_1, ..., y_N\}$: points labels of $\mathcal{X}$
```

1.2.5 . Ordering Points To Identify the Clustering Structure

Ordering Points To Identify the Clustering Structure algorithm (OPTICS) [Ank+99] is based on DBSCAN present in the Section 1.2.4, it enables the identification of clusters with varying densities by employing multiple iterations of the DBSCAN algorithm. Still, it constructs reachability accessibility between each point for evaluating their density compared with the dataset before applying DBSCAN. Clusters are hierarchical, contrary to DBSCAN; if a point does not belong to a cluster, *e.g.*, if $\#\mathcal{N}_\epsilon(x_i) \leq MinPts$, OPTICS adds a new cluster to the hierarchy and analyses the next neighboring points. This difference in processing makes it possible to identify the points at the border of the clusters. OPTICS does not require specifying a density threshold to detect clusters with different densities, instead of DBSCAN with the reachability distance. The process is highlighted in Algorithm 5.

OPTICS has a higher computational time for extensive datasets than DBSCAN as it constructs a hierarchical clustering, and the cluster hierarchy generated by OPTICS is difficult to interpret. As DBSCAN, OPTICS is sensitive to the ϵ and $MinPts$ parametrization.

Algorithm 5 Ordering Points To Identify the Clustering Structure Algorithm.

Input :
Set of points to be clustered, $\mathcal{X}$
Minimum number of points to form a cluster, $MinPts$
Radius ϵ, DBSCAN algorithm (in 4), $DBSCAN$

Procedure :

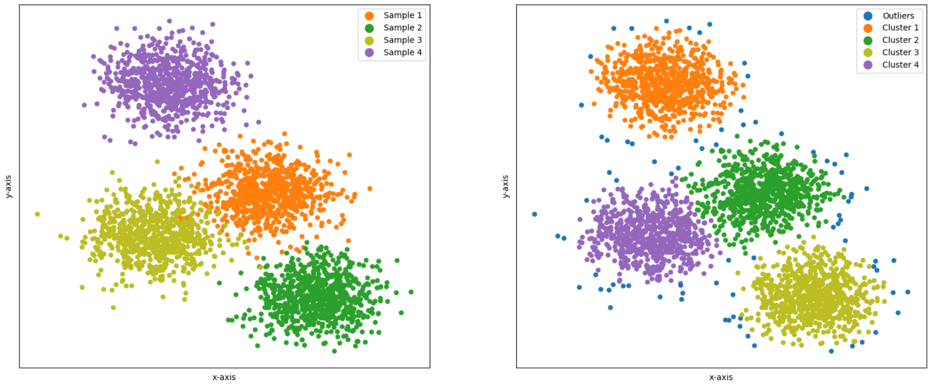
- Computation of Cores distances $d_{core}(x_i)$ for all x_i of $\mathcal{X}$: distance between x_i and its $MinPts$-nearest neighbor if $\#\mathcal{N}_\epsilon(x_i) \geq MinPts$ else undefined
- Computation of the reachability distance between a point x_i and all others points x_n of $\mathcal{X}$: $d_{ReachDist}(x_i, x_n) = \max\{d_{core}(x_n), dist(x_i, x_n)\}$ if $\#\mathcal{N}_\epsilon(x_i) \geq MinPts$ else undefined
- Sorting of points according to their distances: from more dense to least dense
- Cluster detection with DBSCAN from the list and their reachability-distances
- Reachability-plot construction and cluster extraction according to a pruning threshold

Output : $\mathcal{Y} = \{y_1, ..., y_N\}$: points labels of $\mathcal{X}$

1.2.6 . Hierarchical Density-Based Spatial Clustering of Applications with Noise

Hierarchical Density-Based Spatial Clustering of Applications with Noise algorithm (HDBSCAN) [CMS13] is a hierarchical version of DBSCAN. As previously explained, in DBSCAN and OPTICS, the key parameters $MinPts$ and ϵ must be estimated to properly ensure good clustering performance. To overcome these limitations, HDBSCAN presented in Algorithm 6 relies on a hierarchical approach that omits the crucial ϵ parameter by providing dendrograms for all DBSCAN clustering solutions. Then, the “best” value of ϵ is chosen thanks to an optimization over the trees.

Considering the DBSCAN application in Figure 1.8 and applying HDBSCAN, HDBSCAN was able to identify the 4 clusters without having to specify the ϵ parameter as shown in the results in Figure 1.9b.



(a) Simulated observations with colors identifying true labels.

(b) HDBSCAN outputs with colors identifying clusters.

Figure 1.9: HDBSCAN application on a simulated dataset.

Algorithm 6 Hierarchical Density-Based Spatial Clustering of Applications with Noise Algorithm.

Input :
Set of points to be clustered, $\mathcal{X}$
Minimum number of points to form a cluster, $MinPts$
Distance, $dist$, DBSCAN algorithm (in 4)

Procedure :

- Distances computation:
  - Computation of Cores distances $d_{core}(x_i)$ for all x_i of $\mathcal{X}$: distance between x_i and its $MinPts$-nearest neighbor
  - Computation of Mutual Reachability distances between all points: $dist_{mr}(x_i, x_n) = \max\{d_{core}(x_i), d_{core}(x_n), dist(x_i, x_n)\}$
- Mutual Reachability Graph construction MG_{MinPts} with edges are the Mutual Reachability distances, the vertices the points, and d_{core} the weights of the corresponding points
- Minimum spanning tree construction MG^*_{MinPts} by iteratively deleting edges in decreasing order of weight in MG_{MinPts}
- Cluster initialization with DBSCAN from MG^*_{MinPts}
- Condensed tree construction from DBSCAN clusters by iteratively merging the clusters according to their densities
- Cluster selection: Parses the tree top-down using $MinPts$ as a threshold to prune the tree

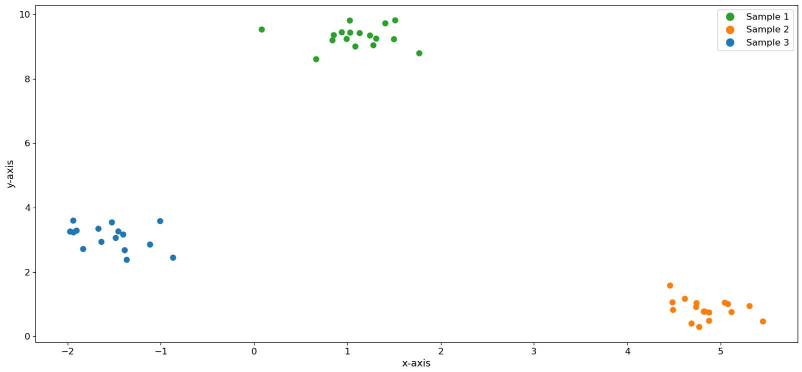
Output : $\mathcal{Y} = \{y_1, ..., y_N\}$: points labels of $\mathcal{X}$

	K-means	**HAC**	**GMM**	**DBSCAN**	**OPTICS**	**HDBSCAN**
Need to specify the number of clusters to find	Yes	Yes	Yes	***No***	***No***	***No***
Outliers detection	No	No	No	***Yes***	***Yes***	***Yes***
Clusters shape	Spherical	Depends on where the aggregations are stopped	Spherical Elliptical	***Heterogeneous***	***Heterogeneous***	***Heterogeneous***
Cluster densities	Homogeneous	Depends on where the aggregations are stopped	Homogeneous	Homogeneous	***Heterogeneous***	***Heterogeneous***
Dataset size	***Small to huge***	Small to medium	Small to medium	***Small to huge***	Small to medium	***Small to huge***
Parametrization	***Simple***	Medium (requires a dendrogram pruning step to stop aggregation and fixed a threshold could be difficult for large dataset)	***Simple***	Medium (requires an ϵ parameters)	Medium (strongly influences algorithmic complexities)	***Simple***
Interpretation	***Simple***	Medium	***Simple***	***Simple***	Medium	***Simple***

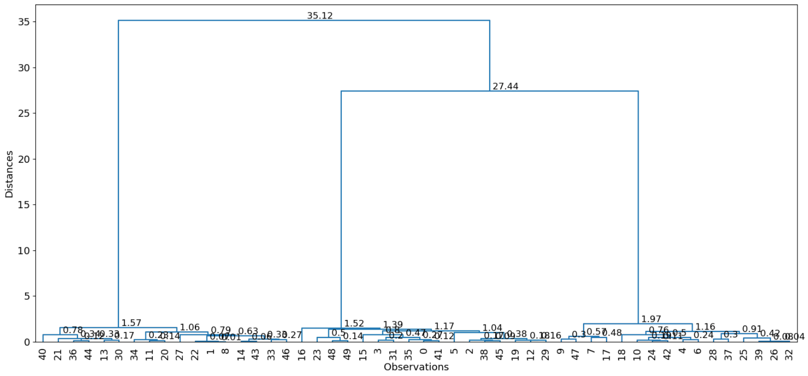
Table 1.2: Clustering algorithms comparisons.

1.3 . Unsupervised metrics to prune a dendrogram

A dendrogram represents the aggregation at each step of the hierarchical agglomerative clustering. As previously explained, pruning a dendrogram is a meaningful challenge; prematurely stopping merges risks, splitting observations from the same class into several groups. Conversely, stopping the mergers late can group observations from several classes into one group. A random dataset was created grouping three samples, as shown in Figure 1.10a, and hierarchical agglomerative clustering was applied using Euclidean distances. The results are displayed in Figure 1.10b; the values represent the distance between each group at each step of the Hierarchical Agglomerative Clustering.



(a) Simulated observations.

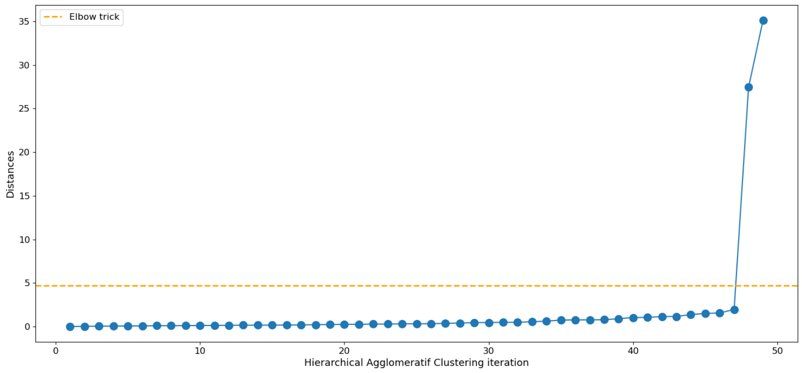


(b) Hierarchical Agglomerative Clustering outputs with each line identifies the results of an unsupervised metric.

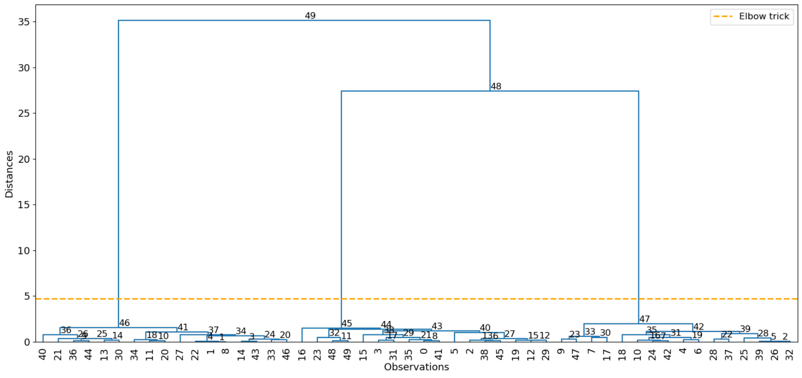
Figure 1.10: Illustration of Hierarchical Agglomerative Clustering.

1.3.1 . Gap Score

The Gap Score (also called the Elbow trick) is based on the analysis distances to detect the breaking point to stop fusion. Let's consider $X = \{x_1, x_2, ..., x_N\}$ a dataset, and a partitioning algorithm giving K clusters such as $\{C_1, C_2..., C_K\}$ with C_i representing the indices of the observations in cluster i and $\#C_i = N_i$. The values are sorted as shown in Figure 1.11a; the orange line represents the breakpoint. This point allows you to stop the aggregations by identifying the iteration number on Figure 1.11b. The agglomeration distances between several observations of the same class are supposed to be small. In contrast, the difference between two observations belonging to two distinct classes will be higher..



(a) Sorted distances.



(b) Hierarchical Agglomerative Clustering outputs with the iteration number displayed.

Figure 1.11: Elbow trick application.

1.3.2 . Silhouette Score

Silhouette Score [Rou87] evaluates each group compared to the entire clustering results using intra-cluster and inter-cluster variances. Let's consider $X = \{x_1, x_2, ..., x_N\}$ a dataset, and a partitioning algorithm giving K clusters such as $\{C_1, C_2..., C_K\}$ with C_i representing the indices of the observations in cluster i and $\#C_i = N_i$; the silhouette score is given by:

$$S_{SIL} = \frac{1}{K} \sum_{i=1}^{N} sil(x_i), \tag{1.14}$$

with $sil(x_i)$ the average of the silhouette coefficient for all points given by:

$$sil(x_i) = \frac{V_{inter}(x_i) - V_{intra}(x_i)}{\max(V_{inter}(x_i), V_{intra}(x_i))}, \tag{1.15}$$

with V_{inter} the inter-cluster variance defined as the average distance between a point to each point from the nearest cluster and V_{intra} the intra-cluster variance defined as the average distance between each point in a cluster.

$$V_{intra} = \sum_{i=1}^{k} \sum_{x_i \in C_i} \|x - \overline{x_i}\|^2 \text{ and } V_{inter} = \sum_{i=1}^{k} n_i \|\overline{x_i} - \overline{x}\|^2, \tag{1.16}$$

with n_i the number of point in the cluster i, $\overline{x_i}$ is the average of the cluster i, $\overline{x}$ the global average of all samples, c_i the cluster i.

The silhouette score values range from -1 to 1. -1 indicates inadequate cluster separation, a value close to 0 is a cluster overlapping, and 1 is a perfect separation, i.e., all the points of a cluster are close to each other and far from the points of the other clusters.

Figure 1.12 represents the Silhouette Score values at each iteration of Hierarchical Agglomerative Clustering from Figure 1.10. By selecting the maximum value, three samples are identified in the dataset.

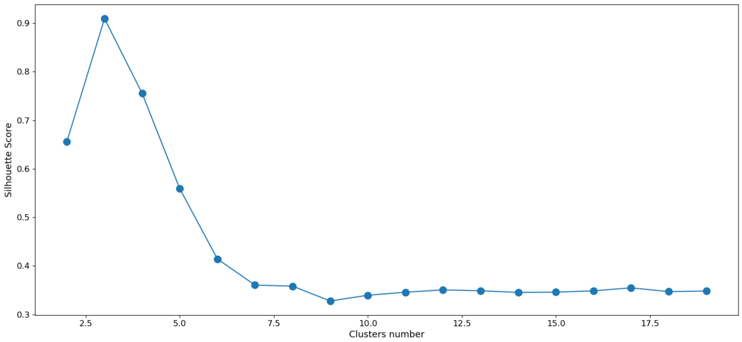


Figure 1.12: Silhouette Scores values computed at each step of the Hierarchical Agglomerative Clustering.

1.3.3 . Calinski-Harabsz Score

Calinski-Harabsz Score [CH74] measures the cluster dispersion by calculating the intra-cluster and inter-cluster variances ratio. Let's consider $X = \{x_1, x_2, ..., x_N\}$ a dataset, and a partitioning algorithm giving K clusters such as $\{C_1, C_2..., C_K\}$ with C_i representing the indices of the observations in cluster i and $\#C_i = N_i$; the silhouette score is given by:

$$S_{CH} = \frac{(N-K)}{(K-1)} . \frac{V_{inter}}{V_{intra}}, \tag{1.17}$$

with V_{inter} the inter-cluster variance defined as the average distance between a point to each point from the nearest cluster and V_{intra} the intra-cluster variance defined as the average distance between each point in a cluster defined in Equation (1.16).

The Calinski-Harabsz score values range between 0 and $+\infty$. A value close to 0 indicates an inadequate assignation of the point to the clusters.

Figure 1.13 represents the Calinski-Harabsz Score values at each iteration of Hierarchical Agglomerative Clustering from Figure 1.10. By selecting the maximum value, three samples are identified in the dataset.

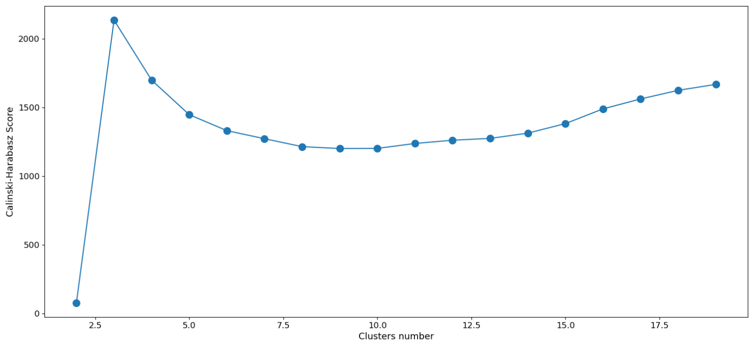


Figure 1.13: Calinski-Harabsz Score values computed at each step of the Hierarchical Agglomerative Clustering.

1.3.4 . Davies Bouldin Score

Davies Bouldin Score [DB79] also measures the cluster dispersion by analyzing the intra-cluster and inter-cluster variance. Let's consider $X = \{x_1, x_2, ..., x_N\}$ a dataset, and a partitioning algorithm giving K clusters such as $\{C_1, C_2 ..., C_K\}$ with C_i representing the indices of the observations in cluster i and $\#C_i = N_i$; the silhouette score is given by:

$$S_{DB} = \frac{1}{K} \sum_{i=1}^{K} \max_{i \neq j'} \left(\frac{V_{intra_i} + V_{intra_j}}{V_{inter_{i,j}}} \right), \tag{1.18}$$

with V_{inter} the inter-cluster variance defined as the average distance between a point to each point from the nearest cluster and V_{intra} the intra-cluster variance defined as the average distance between each point in a cluster defined in Equation (1.16).

The Davies Bouldin Score values range between 0 and $+\infty$. A value close to 0 indicates a good assignation of the point to the clusters, while a high value means the points are misassigned.

Figure 1.14 represents the Davies Bouldin Score values at each iteration of Hierarchical Agglomerative Clustering from Figure 1.10. By selecting the minimal value, three samples are identified in the dataset.

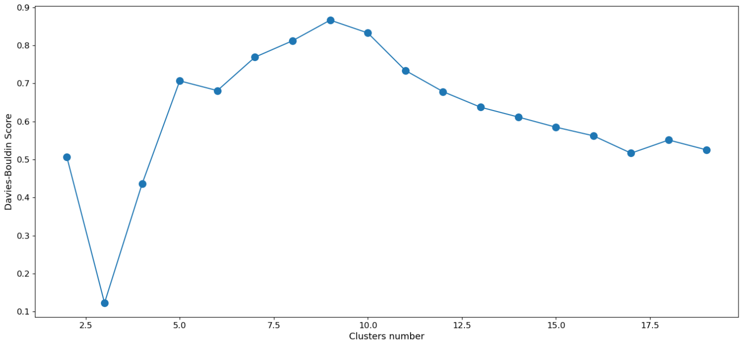


Figure 1.14: Davies Bouldin Score values computed at each step of the Hierarchical Agglomerative Clustering.

1.4 . Statistical test

1.4.1 . Student test

The Student test [Stu08] is a parametric test used to determine if two independent samples $X = \{x_1, x_2, ..., x_N\}$ and $Y = \{y_1, y_2, ..., y_M\}$ with unequal sizes and variances come from the same dataset. The test compares their averages to determine if they are significantly different. The test can be written as follows:

$$\begin{cases} H_0 : E(X) = E(Y) \\ H_1 : E(X) \neq E(Y) \end{cases} \tag{1.19}$$

under H_0, the null hypothesis indicates that the two distributions have expected identical average values, meaning the two samples come from the same dataset. The confidence level is commonly set at 95%, meaning H_0 is accepted with an error risk α lower than 5%. From the Student theorem, the statistic is defined as follows:

$$t_{std} = \frac{\overline{X} - \overline{Y}}{\sqrt{\frac{S_X^2}{N} + \frac{S_Y^2}{M}}} \sim \mathcal{T}(\nu), \tag{1.20}$$

with N and M are the samples sizes of X and Y, ν is the degree of freedom of the distribution, $\overline{X}$ and $\overline{Y}$ are the empirical average of the sample X and Y given by:

$$\overline{X} = \frac{1}{N}\sum_{i=1}^{N} x_i \text{ and } \overline{Y} = \frac{1}{M}\sum_{j=1}^{M} y_j. \qquad (1.21)$$

S_X^2 and S_Y^2 are the empirical variances of X and Y given by:

$$S_X^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{X}) \text{ and } S_Y^2 = \frac{1}{M}\sum_{j=1}^{M}(x_j - \overline{Y}). \qquad (1.22)$$

The test statistic t_{std} get from Equation (1.20) is compared to the critical value t_α associated with a confidence level and the degree of freedom ν (or named degree of normality) [Kru13] to conclude about the test. The Student's test can be approximated by a Normal distribution when the degree of freedom is infinite.

Figure 1.15 represents an example of the Student test with an n degree of freedom for a bilateral test. If the value of t_{std} is in the acceptance zone, the hypothesis null is accepted, meaning the two samples come from the same dataset.

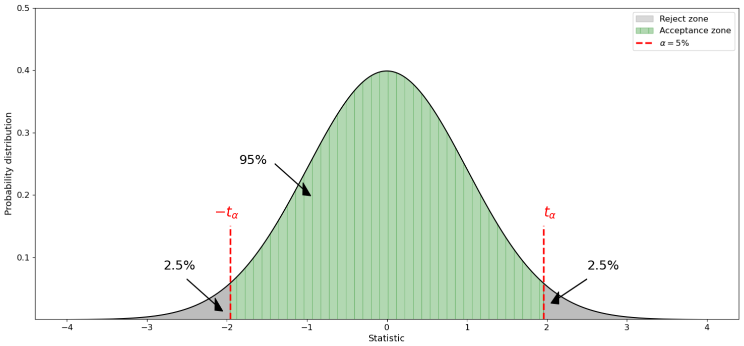


Figure 1.15: Decision zone according to statistical value.

1.4.2 . Kolmogorov-Smirnov test

The Kolmogorov-Smirnov test[Mas51] is a non-parametric test that compares the cumulative, continuous distribution functions of a sample to determine if this sample follows a specific distribution. The test also compares the cumulative distribution functions $F_X(x)$ and $F_Y(y)$ of two samples $X = \{x_1, x_2..., x_N\}$ and $Y = \{y_1, y_2..., y_M\}$ with unequal sizes and variances come from the same distribution. The test is based on analyzing the difference between the empirical distribution functions of the two samples. The test can be written as follows:

$$\begin{cases} H_0 : F(x) = G(x) \\ H_1 : F(x) \neq G(x) \end{cases} \tag{1.23}$$

under H_0, the null hypothesis indicates that the two distributions come from the same sample. The confidence level is fixed, similar to the student test. From the Kolmogorov-Smirnov theorem, one have the Kolmogorov-Smirnov's statistic value:

$$t_{ks} = \max\{\|F_X(x) - F_Y(y)\|\}. \tag{1.24}$$

With $F_X(x)$, $F_Y(y)$, the cumulative distribution function of two samples given by:

$$\begin{aligned} F_X(x) &= \mathbb{P}(X \leq x) & F_Y(y) &= \mathbb{P}(Y \leq y) \\ &= \sum_{x_i \leq x} p_i & &= \sum_{y_j \leq y} q_j, \end{aligned} \tag{1.25}$$

with p_i and q_j the probabilities X takes x_i values and Y take y_j values. The critical value t_α is obtained from the distribution table of Kolmogorov-Smirnov and analyzed similarly to the Student-test.

1.4.3 . Epps-Singleton test

The Epps-Singleton test [ES86] is based on the Kolmogorov-Smirnov test that compares the characteristic empirical function $\mu(x)$ and $\nu(y)$ of two samples $X = \{x_1, x_2..., x_N\}$ and $Y = \{y_1, y_2..., y_M\}$ to determine if they come from the same distribution. The Epps-Singleton test is more robust to outliers than the Kolmogorov-Smirnov test and makes no assumptions about data distribution continuity. Unlike the Kolmogorov-Smirnov test, it can be used when the distributions can be asymmetric or multi-modal. The test can be written as follows:

$$\begin{cases} H_0 : \mu(x) = \nu(y) \\ H_1 : \mu(x) \neq \nu(y). \end{cases} \tag{1.26}$$

Under H_0, the null hypothesis indicates that the two distributions come from the same sample. The confidence level is fixed, similar to the student test. From the Epps-Singleton theorem, one have the Epps-Singleton's statistic value:

$$t_{epps} = \int_{-\infty}^{\infty} [\mu(x) - \nu(y)]^2 \, dF_X(x) \, dF_Y(y). \tag{1.27}$$

Where the empirical characteristics functions $\mu(x)$ and $\nu(y)$ are given by:

$$\mu(x) = \exp{(ix\overline{X} + \frac{1}{2}x^2 S_X^2)} \text{ and } \nu(y) = \exp{(iy\overline{Y} + \frac{1}{2}y^2 S_Y^2)}, \tag{1.28}$$

with $\overline{X}$, $\overline{Y}$ and S_Y^2, $2S_Y^2$ the empirical average and variances of X and Y give by Equations (1.21) and (1.22). $F_X(t)$ and $F_Y(t)$ are the distribution function get from Equation (1.25). N and M are the samples sizes of X and Y: $N = \#X$, $M = \#Y$. The critical value t_α is obtained from the distribution table of Epps-Singleton and analyzed similarly to the Student-test.

1.5 . Supervised metrics for evaluating methods

1.5.1 . Adjusted Rand index

Adjusted Rand Index (ARI) [HA85] quantifies the similarity between two sets of classes $E = \{e_i | i = 1, ..., P\}$ and $F = \{f_j | j = 1, ..., Q\}$ from Ω with $N = \#\Omega$, based on the sum squares of the numbers of points assigned to E and F. The score value is given by:

$$M_{ari} = \frac{\sum_{ij} \binom{c_{ij}}{2} - [\sum_i \binom{e_i}{2} \sum_j \binom{f_j}{2}] / \binom{c}{2}}{\frac{1}{2}[\sum_i \binom{e_i}{2} + \sum_j \binom{f_j}{2}] - [\sum_i \binom{e_i}{2} \sum_j \binom{f_j}{2}] / \binom{c}{2}}. \tag{1.29}$$

From C the contingency table between E and F, one have:

$$c = \sum_{i=1}^{P} \sum_{j=1}^{Q} c_{ij}, \tag{1.30}$$

wheree_i is the row sums: $\sum_{i=1}^{P} c_{ij}$ and f_j the columns sums: $\sum_{j=1}^{Q} c_{ij}$. ARI values range from -1 to 1. -1 indicates a complete mismatch between two partitions, whereas one indicates a perfect match.

1.5.2 . Homogeneity score

Homogeneity score [RH07] measures the degree of similarity between classes from two sets of classes $E = \{e_i | i = 1, ..., P\}$ and $F = \{f_j | j = 1, ..., Q\}$ from Ω, with $N = \#\Omega$. Homogeneity is based on analyzing the entropy between E and F. The measured value is given by:

$$M_{hm} = 1 - \frac{entropy(E|F)}{entropy(E)}, \tag{1.31}$$

with C the contingency table between E and F, and the $entropy(E)$ is given by:

$$entropy(E) = -\sum_{i=1}^{P} \frac{\sum_{j=1}^{Q} c_{ij}}{N} \log \frac{\sum_{j=1}^{Q} c_{ij}}{N}, \tag{1.32}$$

and $entropy(E|F)$ the entropy of E conditionally to F is given by:

$$entropy(E|F) = -\sum_{j=1}^{Q} \sum_{i=1}^{P} \frac{c_{ij}}{N} \log \frac{c_{ij}}{\sum_{i=1}^{P} c_{ij}} \tag{1.33}$$

Homogeneity values range from 0 to 1; 1 indicates that the class points are not mixed in the same group, whereas 0 indicates that some groups contain points belonging to several classes.

1.5.3 . Completeness score

Completeness [RH07] measures the spreading points across multiple groups between two sets of classes $E = \{e_i | i = 1, ..., P\}$ and $F = \{f_j | j = 1, ..., Q\}$ from Ω, with $N\#\Omega$. Completeness is symmetrical to Homogeneity. The value is given by:

$$M_{cp} = 1 - \frac{entropy(F|E)}{entropy(F)}, \tag{1.34}$$

with C the contingency table between E and F and where the $entropy(F)$ is given by:

$$entropy(F) = -\sum_{j=1}^{Q} \frac{\sum_{i=1}^{P} c_{ij}}{N} \log \frac{\sum_{i=1}^{P} c_{ij}}{N}. \tag{1.35}$$

The entropy of F conditionally to E, $entropy(F|E)$ is given by:

$$entropy(F|E) = -\sum_{i=1}^{P} \sum_{j=1}^{Q} \frac{c_{ij}}{N} \log \frac{c_{ij}}{\sum_{j=1}^{Q} c_{ij}}. \tag{1.36}$$

Completeness values range from 0 to 1; 1 indicates that all points belonging to a class are grouped into one class, whereas 0 indicates that the point of one class is spread across multiple groups.

1.6 . Kernel Density Estimation

Kernel Density Estimator [Ros56; Par62] (KDE) is a non-parametric method to estimate the distribution parameters and the density function f_X from a random variable based on kernels considering each point from a sample $X = \{x_1, x_2..., x_N\} \in \mathbb{R}$ as a density peak and built a density function around them to smooths a curve to estimate the overall density distribution. Kernel estimation does not make assumptions about the shape of the underlying distribution of the data. Ones kernel density estimator of f_X at point x obtained from Parzen [Par62] is given by:

$$\begin{aligned}\hat{f}_h(x) &= \frac{1}{Nh}\sum_{i=1}^{N} K_h\left(x, x_i\right) \\ &= \frac{1}{Nh}\sum_{i=1}^{N} K\Big(\frac{x - x_i}{h}\Big),\end{aligned} \tag{1.37}$$

with N the data size, K the kernel, and h the smoothing parameter (bandwidth). The KDE mainly depends on kernel choice and bandwidth. The bandwidth parameter is essential and greatly influences the results by fixing a compromise between the estimate's precision and the probability density's smoothing. A low value undersmoothes the data, and the estimated curve will have several peaks, while a high value could mask the data characteristics. Its value is fixed according to the kernel used and the data structure. The Gaussian Kernel is the most used, but other kernels can also be used as the Uniform, Epanechnikov, or Laplace kernel.

Figure 1.16 shows an application of Kernel density estimation on a dataset. The data plotted in blue with the histogram shows the presence of 3 Gaussians with different parameters; the Gaussian kernel, therefore, appears to be the best choice for estimating the probability density of this example.

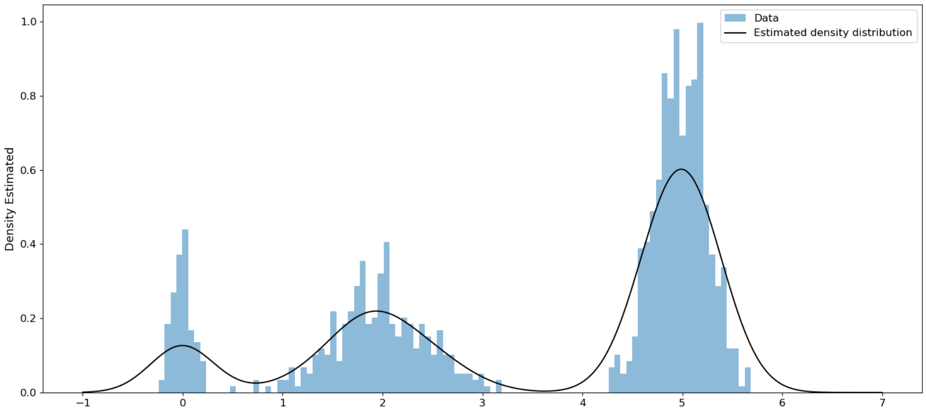


Figure 1.16: Gaussian Kernel Density Estimation application.

The Gaussian kernel [BA97] can be written as:

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}, \tag{1.38}$$

with $X \sim \mathcal{N}(\mu, \sigma^2)$. Combining Equations (1.38) and (1.37) gives:

$$\begin{aligned} \hat{f}_X(x) &= \frac{1}{N} \sum_{i=1}^{N} \phi_h \left(x - x_i\right) \\ &= \frac{1}{Nh\sqrt{2\pi}} \sum_{i=1}^{N} e^{-\frac{1}{2}\left(\frac{|x-x_i|}{h}\right)^2}, \end{aligned} \tag{1.39}$$

with ϕ_h standard normal density function, and the optimal bandwidth h can be obtained from several methods [Sco15; Sil86; SJ91; Hal+91]. From Scott's method, the bandwidth is given by:

$$h = 1.06 \hat{\sigma} N^{-1/5}. \tag{1.40}$$

This chapter presented the mathematical methods necessary to understand the approaches developed for the RADAR Reconnaissance Process, illustrated with examples.

2 - Data, modeling, and simulation of RADAR signals

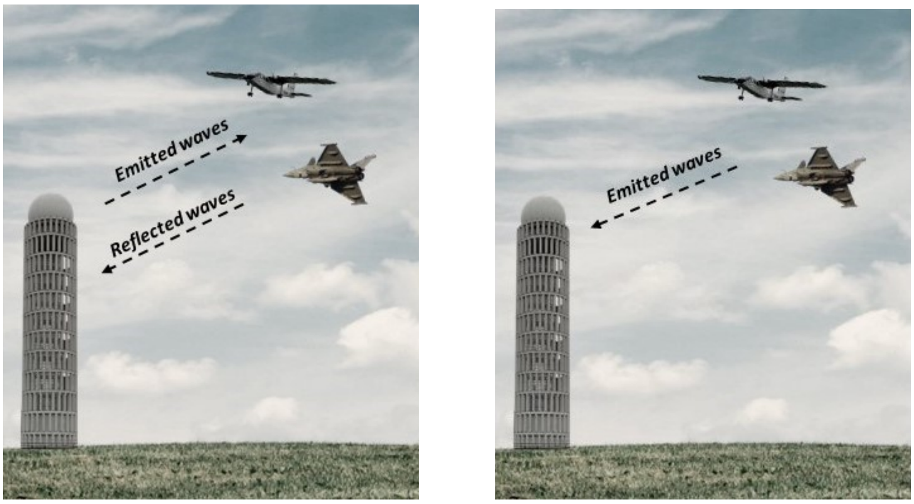
Before presenting the approaches developed for the RADAR Recognition Process, this chapter presents the fundamental principles necessary for understanding the functioning of RADAR systems, specifically for passive RADAR. RADAR emitters' data is crucial in many fields, such as object recognition, anomaly detection, and autonomous decision-making. Collecting real RADAR emitters' data can be expensive and limited in quantity, which makes simulated RADAR emitters' data extremely valuable. Simulated data must accurately represent the characteristics of the RADAR emitters' system. Advanced modeling and simulation methods are required to realistically reproduce RADAR emitters' signals, considering the effects of noise, interference, and environmental characteristics. The proposed methodologies were developed and evaluated using several data simulators: a simulator provided by Avantix involved in the Ph.D thesis that declined to communicate any related information due to confidentiality, and a second developed during the research, which will include more comprehensive details. The latter will be detailed to better understand the dataset composition. The signals used are simulated using the characteristics of emitter classes. Their characteristics are structured to facilitate their understanding and use through our approaches. A section will detail all the processes implemented to structure the data, accompanied by examples of simulated signals. The last section presents an overview of the existing methods to deinterleave RADAR signals and identify emitters.

2.1 . Radio Detection and Ranging

2.1.1 . Basic principles

RADAR is an acronym for Radio Detection And Ranging, designating a remote-sensing device that locates and detects objects using electromagnetic waves. The device is based on radiating electromagnetic waves propagating through space until they reach an object and receive the reflected waves. The time measurement of the electromagnetic waves can determine the distance to the object or its speed. The origins of RADAR can be traced to 1890 when James Clerk Maxwell and Heinrich Rudolf Hertz developed the electromagnetic light theory. However, it was in the 1930s that Lawrence A. Hyland made the first successful detection of an aircraft. The Second World War accelerated the development of RADAR equipment and its sophistication to help protect the Allied fleet against German U-boat attacks, leading to new models.

RADAR's development has been the most significant advance in sensing remote objects since the invention of the telescope in 1608. RADAR has become a vital issue in the field of electronic intelligence [Sch80; Ner06]. Since then, they have been used in many fields besides military applications, such as meteorology, air traffic monitoring, remote sensing of the environment, search and rescue at sea, space, or archaeology. Since this invention, RADAR emitters have provided a breakthrough in Electronic Intelligence, specifically for extracting information to respond appropriately to hostile situations [Wil06; Ada03].



(a) Monostatic active RADAR.

(b) Monostatic Passive RADAR.

Figure 2.1: RADAR operating architecture.

Among the different technologies, the distinction between active RADARs and passive RADARs is made due to their functioning [Sko08; Wil06; Sch99], as illustrated in Figure 2.1. An active RADAR, as shown in Figure 2.1a, generates a waveform with a modulator for transmitting toward a target. These waves propagate through the air and contact the target with the emitter's powerful radio frequency pulse. The reflected waves are detected by a receptor that measures the time difference between the emission of the waves and the collection of their echoes. It amplifies and demodulates the received signals to convert them into numerical signals. Algorithms process the electromagnetic signals received by the receptor to extract information such as the direction, speed, size, and shape of the detected object to be presented in a graphical form, such as in the plan position indicator. These results guide ESM operators in decision-making and military engagement. Conversely, a passive RADAR, as shown in Figure 2.3a, does not emit electromagnetic waves; it detects electromagnetic signals already present in the environment, such as enemy radar emissions, communications signals, or even natural emissions. It allows discreet surveillance.

After intercepting a signal, the operators receive a raw signal as illustrated in Figure 2.2, representing a simulated signal grouping together three pulses. The analog signal is converted into digital with an analog/digital conversion device to extract information.

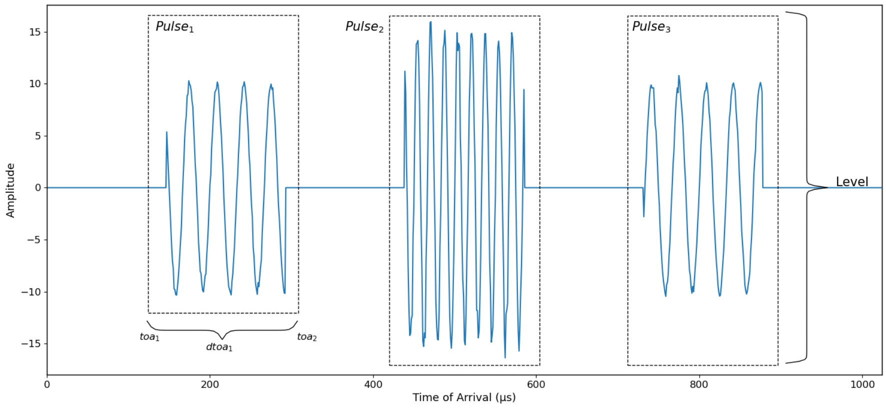


Figure 2.2: Example of three pulses from an intercepted signal.

2.1.2 . Emitter classification system

A wide variety of emitters leads to modifying the operation described above. A classification according to their mode, *e.g.*, their function, can be processed to distinguish them. Some emitters monitor and control air traffic and can detect the position, the trajectory, or the speed over a wide area of a target, such as aircraft or missiles. Weather emitters were initially used for military purposes to locate precipitation, calculate its movement, and determine its type to separate clutter from precipitation and improve aerial surveillance. Other RADAR emitters aim to track targets and their trajectory, or the Surface Movement Radar enables the detection of vehicles and aircraft on the tarmac and runways, etc. Their operation mode determines the orientation of their range, radial velocity, angular direction, or how to cover the listening perimeter. Various other nomenclatures exist to categorize emitters:

- RADAR according to their functionality:
    - Primary RADAR: emits microwave signals reflected by a target and analyzes the reflected part of its signal.

  - Secondary RADAR: is used with a primary RADAR for acquiring additional target information: it uses signals reflected from a transponder.

- RADAR according to their emitted pulses:

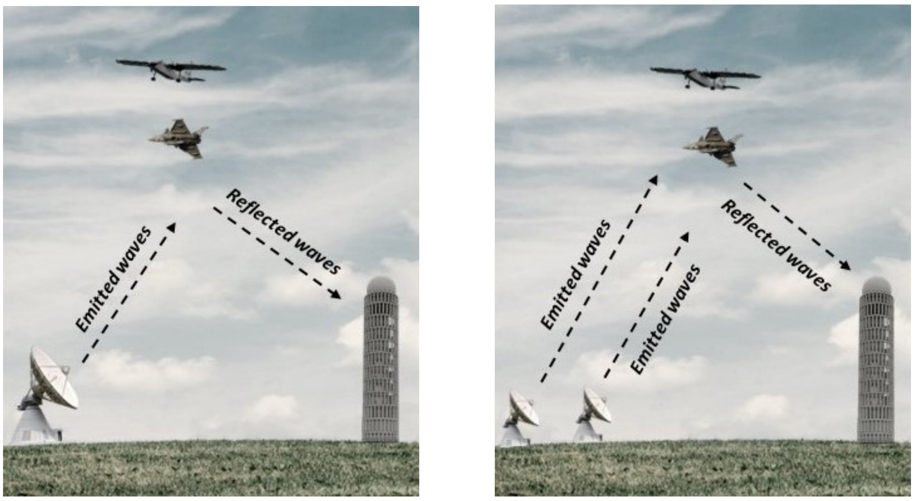
  - Pulses waves RADAR: RADAR emitters transmit short and powerful pulses and listen to the environment to intercept the echo returned by the target.
    - Intra-pulse modulation: the frequency, phase, or amplitude of the electromagnetic wave pulse within each pulse is modified.
    - Non-intrapulse: the transmitted pulses are short and non-modulated.
  - Continuous Wave Radar: Continuous wave radars generate a continuous microwave signal. The reflected signal is received and processed:
    - Continuous waves RADAR (CW-RADAR): continuously transmits a microwave signal and is unmodulated; the echo is therefore received and processed continuously.
    - Frequency Modulated Continuous Wave RADAR (FM-CW RADAR): using frequency-modulated continuous waves. The echo is analyzed after being reflected, considering the frequency variation between the transmitted signal and the reflected signal.
  - Pulses Doppler RADAR: RADAR emitters measure the target's speed by detecting the frequency shift of reflected signals in clutter. They are frequently used for monitoring air traffic, detecting the speed of vehicles on roads, and detecting precipitation in meteorology.

- RADAR according to the antennas used:

  - Monostatic RADAR: uses the same antenna for transmitting and receiving radar signals, illustrated in Figure 2.1a.
  - Bistatic RADAR: RADAR's receiver and transmitter do not use the same antenna as illustrated in Figure 2.3a; it uses two separate antennas, one for transmission and one for reception, which are placed at different positions.
  - Multistatic RADAR: uses multiple receiving stations, with separate antennas, to detect a target, as illustrated in Figure 2.3b. They have a more excellent detection range, better spatial resolution, and improved detection ability than monostatic and bistatic RADAR emitters.

- RADAR according to their mode:
  - Imaging radars: generates an image of the observed object or area.
  - Tracking RADAR emitters: tracks the position and movements of a moving object, such as a plane, ship, or tracking enemy missiles or aircraft.

Emitters can be classified through multiple categories based on their application, sophistication, or resolution. In the following, the focus will be made on passive sensors, which, through language misuse, are called emitters.



(a) Bistatic active RADAR.

(b) Multistatic active RADAR.

Figure 2.3: RADAR according to the antennas used.

2.2 . Emitters characteristics

Depending on their mode, emitters have different characteristics:

- **Technical emitter parameters**: Power Transmitted, transmit antenna gain, scan period, or antenna length
- **Pulses parameters**: Frequency, pulse width, level, or pulse repetition interval pattern
- **Receptor characteristics**: spatial distance from the receptor or direction of arrival according to the positions between receptor and emitter.

All these parameters are established according to the field of application of the emitter and its manufacturer. Various models exist, some very different, while others share similar characteristics (e.g., in ports or airports).

When an ESM operator copes with a situation where he must identify the emitter in front of him, he often needs access to all its characteristics and must decide with insignificant information. All these features are unavailable in practice, and the operators base their analyses on the pulse parameters.

All the data presented in this manuscript are simulated and produced by the simulators subsequently presented. For confidentiality considerations, no real data is presented. Models are tuned, parameterized, and validated on simulated data. The complexity of the simulated data represents one of the main challenges in the RADAR processing chain. As part of the collaboration with Avantix, their RADAR signal simulator, developed from their expertise, has been made available.

A second simulator has been developed, based on theory and telecommunication engineering, to provide more public and understandable information on the construction of methodologies. The simulator is designed to mimic the behavior of emitters, with some characteristics being easy to simulate, such as frequency or pulse width, while others necessitate pre-processing, such as the technical specifications of the emitter or the pulse repetition interval, before being able to simulate a RADAR signal.

The next section presents the data developed for the methodology and the experimental procedure to obtain labeled unclassified data.

2.3 . Data modeling

From a database gathering the emitter's characteristics, a signal is simulated according to the user-defined parameters. Multiple emitters with heterogeneous profiles have been simulated to construct the database, and their characteristics are presented in Table 2.1. A limited set of parameters was intentionally selected to represent the emitters; the main reason was to use a few reliable and well-estimated characteristics when intercepting a signal to build robust algorithms. These choices will be more detailed in Chapter 4. Emitters are described by the following parameters:

- **Technical parameters**: Power transmitted, antenna width, the distance between the emitter and the receptor, and the scan period.

- **Pulses parameters**: frequency (F, f_n), pulse width, (PW, pw_n) and pulse repetition interval (PRI, pri_n).

A few technical parameters describe each emitter, such as the transmitted power, the antenna width, the distance between the emitter and the receiver, and the scan period. There is a wide variety of emitters. Some may have elementary parameters, such as emitting on a single frequency, and others sophisticated pulse characteristics, *e.g.*, agility in frequency, multiple pulse repetition of intervals, or with rarer appearance periods. Generally, an emitter's frequency and pulse width emission pattern are sufficient to discriminate them. When the listening perimeter concerns a port, these characteristics are usually similar and/or identical. The PRI is a very discriminating feature of emitters as it is usually unique and defines the signature of an emitter. Emitters may have various operational modes concerning the pulse repetition period. Emitters commonly transmit within a fixed value interval, contrary to complex emitters, which a random PRI distribution characterizes. As the order of appearance of the PRI is not taken into account, a discrete measure as the sum of a few Dirac masses easily represents the deterministic PRI process:

$$\mu_j = \sum_{n=1}^{N} \alpha_n \delta_{pri_n}, \tag{2.1}$$

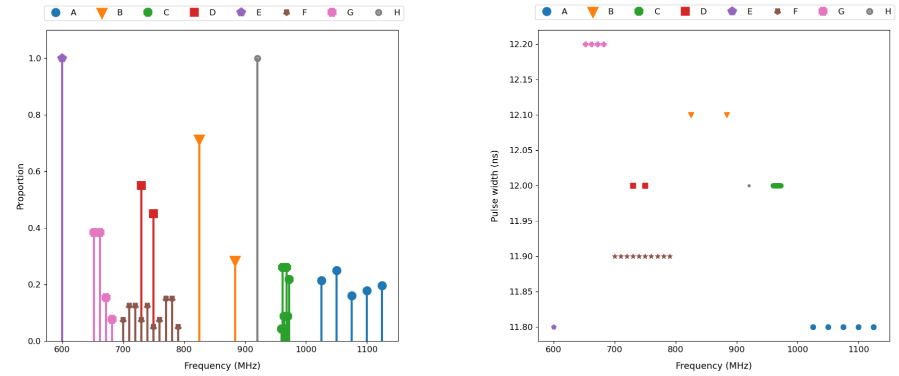
with j the emitter index, N the number of PRIs on which the emitter transmits, α the proportion of the PRI, and δ, the Dirac mass (with $\sum_n \alpha_n = 1$).

Conversely, an emitter transmitting through a random PRI process requires pre-processing as the distribution is not directly representable by a discrete distribution. It is assumed that the PRI distribution oscillates around an average value characterized by the distribution. A Gaussian function can represent the distribution as they are easy to interpret and implement. In addition, Chapter 4 proposes a new representation of the PRI process involving the addition of the PRI values requiring the assumption of Gaussianity for the representation of the PRI of the emitters having a random process:

$$P(x)_j = \frac{1}{\sigma_j \sqrt{2\pi}} \exp\left(-\frac{(x-\mu_j)^2}{2\sigma_j^2}\right), \tag{2.2}$$

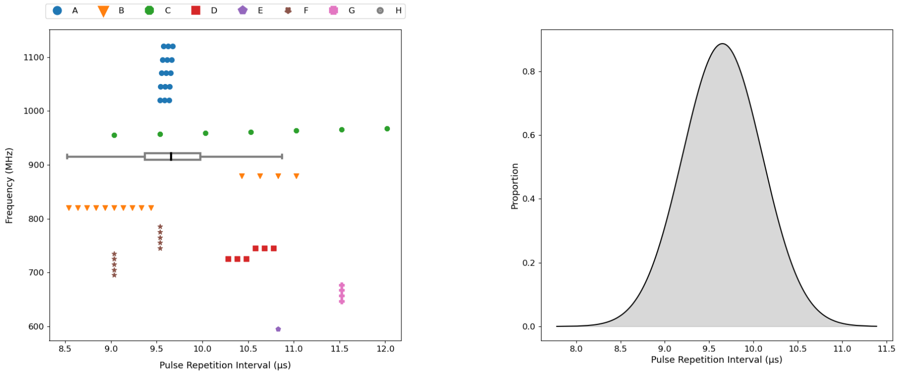
with j the emitter index, μ of the PRI average value, and σ its standard deviation.

The emitters characteristics, presented in Table 2.1, are plotted in Figure 2.4. Each color represents an emitter. Emitter E has elementary characteristics, including single-frequency and pulse-width transmission, contrary to emitter A, which is characterized by 5 frequencies (1025, 1050, 1075, 1100, and 1125 MHz). Emitters D and F share frequencies at 730 and 750 MHz and close pulse widths, complicating their separability. Emitter B has multiple operating modes; it initiates a fast transmission between 8.5 and 9.5 ns with a frequency of 825 MHz. Then, it increases its pulse transmission to 884 Mhz. Emitter H is characterized by a random PRI distribution and represented by a boxplot: the pulses are generated from 9.2 ns to 10.1 ns. The random PRI process of emitter H is generated from the average value of $[9.2, 10, 1]$ and represented in the last plot: $X \sim \mathcal{N}(9.65, 0.45)$.



(a) Frequency.

(b) Frequency x Pulse Width.



(c) Pulse Repetition Interval x Frequency.

(d) Random process of Emitter H.

Figure 2.4: Representation of emitters characteristics in different planes, identifying by color.

Name	**Frequency (MHz)**	**Pulse Width (ns)**	**Pulse Repetition Interval (μs)**	**Power Transmitted**	**Antenna Width**	**Distance**	**Scan Period**
A	1025 1050 1075 1100 1125	11.8	9.5 9.51 9.52 9.53 9.54 9.55 9.56 9.57 9.58 9.59 9.6 9.61 9.62 9.63 9.64	70	3.2	1 055	4 633 301
B	825 884	12.1	8.5 8.6 8.7 8.8 8.9 9 9.1 9.2 9.3 9.4 10.4 10.6 10.8 11	80	3.5	1 002	6 062 663
C	960 962 964 966 968 970 972	12	9 9.5 10 10.5 11 11.5 12	72	2.6	1 104	3 009 738
D	730 750	12	10.25 10.35 10.45 10.55 10.65 10.75	71	2.8	1 054	10 091 029
E	600	11.8	10.8	83	2.2	1 051	7 286 382
F	700 710 720 730 740 750 760 770 780 790	11.9	9 9.5	87	5.7	1 241	1 837 974
G	652 662 672 682	12.2	11.5	82	5.2	1 039	8 007 690
H	920	12	Min: 9.2 Max: 10.1	84	1.6	1 243	7 090 636

Table 2.1: Simulated Emitter characteristics.

2.4 . Simulator

A RADAR simulator was built to test and validate the methodologies developed. The simulator, illustrated in the Algorithm 7 starts by simulating a noise-free raw signal based on the database characteristics in Table 2.1. Then, according to the parameters set by the user, some noise is added to the signal.

Algorithm 7 RADAR Signal simulation.

```
Data : Database with emitters characteristics, $T$.
Parameters :
Threshold detection, $\lambda$
Size, $N$
List of emitter presents, $L$
Splitting coefficient, $\sigma_{split}$
Noise parameters according to frequency, pulse width, and time of arrival, $\sigma = \{\sigma_{freq}, \sigma_{pw}, \sigma_{time}\}$
Initialization : Create an empty array, $D$
Procedure :
    1. Simulation of a raw signal without noise
       For $i$ in $L$ :
           (a) Selection of characteristics of emitter $i$ from $T$
           (b) Construction of the time of arrival of the pulse $toa_n = cumsum(pri_n)$ according to $N, f_n, pw_n$ and $pri_n$
           (c) Computation of the power between the transmitter and the receptor for each pulse: $Pow_r$
           (d) Creation of a signal with all pulses from (a) and (b) and add pulses in $D$
    2. Signal noise
           (a) Adding Gaussian noise $\sigma$ to $D$ on each features
           (b) Truncated pulse width level according to $\sigma_{split}$ and $\lambda$: $S$
Result : $S$, Set of interleaving pulses.
```

2.4.1 . Creating a raw signal

A raw signal is simulated without noise from the database grouping emitter characteristics. The RADAR simulator assumes the receiver is static, has a known detection threshold, and is omnidirectional. The simulator starts by generating pulses from the characteristics entered in the database, with each emitter described by I frequency (f) with $I \in \mathbb{R}_+^*$, J pulse width (pw) with $J \in \mathbb{R}_+^*$ and K pulse repetition of interval (pri) with $K \in \mathbb{R}_+^*$. It reconstructs the emission pattern of each emitter. As a reminder, emitters with a random PRI process are represented by a Gaussian distribution and, therefore, many PRIs. The time of arrival is computed by taking the cumulative sum of the PRI taken by the emitter according to the desired size N of the signal:

$$toa_N = \sum_{n=1}^{N} pri_n. \tag{2.3}$$

The transmission formula of Friis transmission calculates the transmission power between the transmitters and the receptor Pow_r for each pulse and is given by

$$Pow_r = Pow_t + Gn_t + Gn_r - A_t, \tag{2.4}$$

with Pow_t, the power transmitted by the RADAR from the database, Gn_t, the transmitter gain, Gn_r, the receiver gain, and A_t, the propagation loss obtained by:

$$A_t = 20 \log\left(\frac{W_L}{4\pi r}\right), \tag{2.5}$$

with r the distance between the receptor and the transmitter from the database and W_L the wavelength obtained with

$$W_L = \frac{c}{f_n}, \tag{2.6}$$

with c the light speed and f_n the frequency pattern of the RADAR. The directivity of an emitter is a crucial parameter to consider when calculating the power radiated or received by this emitter, as the energy depends on the antenna's orientation and is not distributed uniformly. The energy is powerful when the emitter is facing the target, while the intensity decreases when the emitter rotates.

The simple case of a linear antenna is modeled, and the directivity dir_n is obtained by:

$$dir_n = 10\log_{10}\left(\frac{1}{\text{sin}(\theta)}\text{sin}\left(\frac{ky}{2}\text{sin}(\theta)\right)\right)^2, \tag{2.7}$$

with y the antenna width get from the database, k the wave number, and θ, the antenna orientation, a function depending on the scan period:

$$\theta = \frac{2\pi}{S}toa_n, \tag{2.8}$$

with S, the scan period gets from the database, toa_n the time of arrival get from 2.3. The simulator controls the number of RADAR emitters in the signal and the desired size. A listening system collects the data without information about the environment. The listening signal covers a wide frequency band between emitters, making the Doppler effect neglectable in our analysis.

2.4.2 . Signal noise

The addition of noise in the signal is correlated with the loss of pulse and the poorly estimated and truncated pulse width level. Let us define ζ, the ratio allowing us to consider the low-level pulses:

$$\zeta = (10^{Pow_r/10})^{-1}, \tag{2.9}$$

with Pow_r the power compute from Equation (2.4). Gaussian noise is applied to each feature according to the parameters $\sigma = \{\sigma_{freq}, \sigma_{pw}, \sigma_{time}\}$ set by the user. Let us calculate the noise variances added to the features:

$$\Sigma_{time} = \zeta\sigma_{time}, \Sigma_f = \zeta\sigma_f \text{ and } \Sigma_{pw} = \zeta\sigma_{pw}, \tag{2.10}$$

with these parameters, the features are noisy by adding a Gaussian noise:

$$\begin{aligned}\widetilde{f_n} &= f_n + X, X \sim \mathcal{N}(0, \Sigma_{freq}) \\ \widetilde{pw_n} &= pw_n + Y, Y \sim \mathcal{N}(0, \Sigma_{pw}) \\ \widetilde{toa_n} &= toa_n + Z, Z \sim \mathcal{N}(0, \Sigma_{time})\end{aligned} \tag{2.11}$$

When a signal is poorly estimated or sampled, distortions can appear on the signal: some pulses can be truncated or split. The signal is truncated according to the fixed detection threshold λ, and the truncated pulse width level is modeled from the Poisson distribution according to a parameter σ_{split}. Poisson's distribution describes the probability that an event will occur during an observed time interval, independently of the previous event. Let us define SP as the probability of pulse splitting considering the low level of the pulses with $SP \sim \mathcal{P}(\lambda)$, with

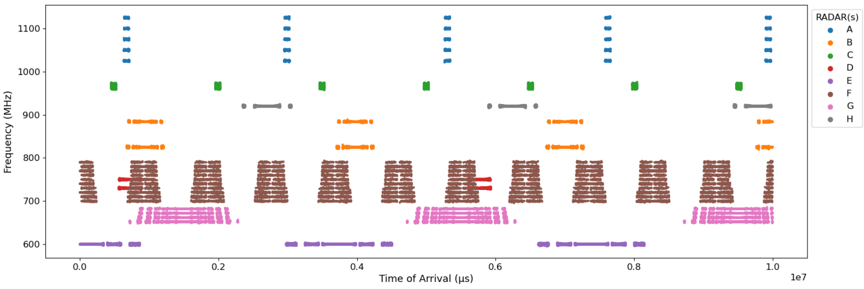
$$\lambda = \frac{\zeta}{\sigma_{split}}. \tag{2.12}$$

The diversity of the simulated signals results in the acquisition of single- or multi-sensor labeled signals, signals with frequency or time modulation, and signals comprising measurement errors, outliers, missing data, or non-Gaussian noise. Figure 2.5 represent examples of signals simulated from the characteristics of the emitters in the Table 2.1.

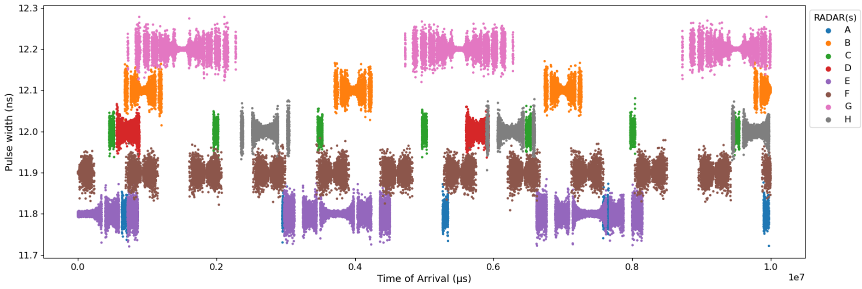
The simulator allows you to generate single-frequency RADAR pulses such as green or violet. They are characterized by a single set of pulses in the (f_n, pw_n) plane in Figure 2.5d. Conversely, RADARs with more complex characteristics transmit on several frequencies, such as blue and brown. Several sets of pulses represent them in this plane. The simultaneity of activity of RADARs can complicate their separability and identification as illustrated in Figure 2.5c. Figure 2.5b shows us a slight spread of the pulse width, indicating an estimation error or the presence of noise in the signal.

An additional feature can be added: the difference of time of arrival (DTOA) δ_n, defined as the interval of time between two successive pulses belonging to the same emitter (in the case of a unique emitter, $\delta_n = toa_n - toa_{n-1}$ which correspond to the PRI of the emitter). Considering a signal mixing pulses from different emitters, two successive pulses are not guaranteed to be transmitted by the same emitter.

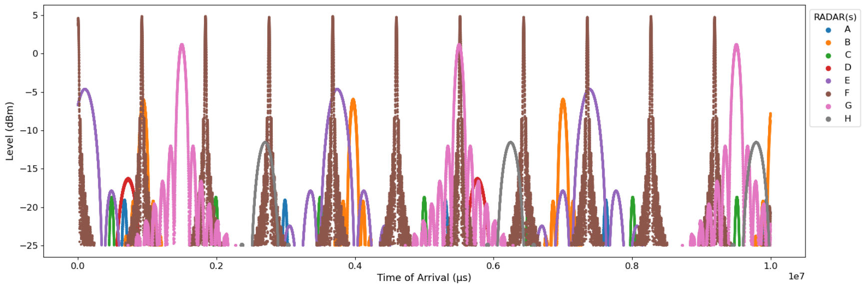
To conclude, this Chapter has introduced the fundamental principles of operating RADAR systems. Emphasis was placed on the interest of using passive systems for our problem. The construction of the data simulator, a central element that enabled the development of the approaches, has been detailed by providing multiple examples of signals for illustration.



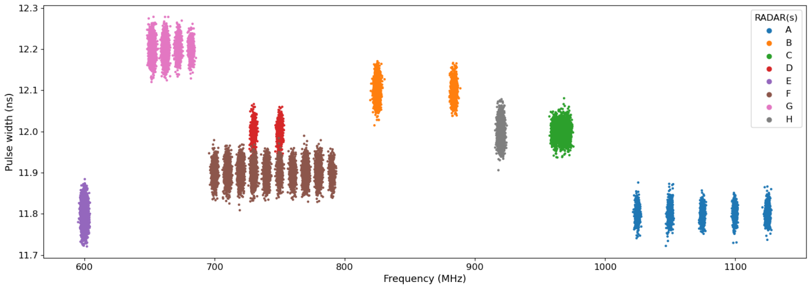
(a) Frequency x time of arrival plane.



(b) Pulse width x time of arrival plane.



(c) Level x time of arrival plane.



(d) Frequency x Pulse width plane.

Figure 2.5: Simulated RADAR signal from all emitters.

2.5 . Overview of the existing methods

This Section offers an overview of the existing RADAR Recognition Process methods. As previously explained in the Introduction, this process comprises two stages; the first consists of deinterleaving a signal by separating and grouping the mixed pulses of an unknown number of emitters before identifying them from a reference database in the second stage. A review of methods will be presented with emphasis on their limitations.

Although these two steps are considered independent in this research, the deinterleave and identification methods are generally done simultaneously by searching the characteristics of the emitter present in the signal from a reference database. Few methods focus solely on the identification of emitters. Although this state-of-the-art will present several identification methods, it mainly references existing methods for deinterleaving a signal. Subsequently, we will distinguish two ways of identifying RADAR in a signal; the first is done during the deinterleaving stage, and the second uses dedicated methods, presented at the end of the chapter.

Generally, this reference database groups a few references [LY19; Din+18; NAA20], describing the waveforms or the pulse repetition of interval (PRI) types, not the RADARs themselves. That is to say, emitter identification methods mainly rely on identifying the waveform or the PRI type from a signal.

In a passive listening system [Sko08; WW85; CR93], an Electronic Support Measure system [DH82; Wil06; Sch99] intercepts signals containing interleaved time domain pulses from an unknown number of emitters. The challenge is providing a rapid response to ESM operators by analyzing this signal to help them make decisions by dealing with an increasingly dense electromagnetic environment [Sch86; Ada14]; it is necessary to adapt and automate decision-making techniques.

Pulse description words (PDW) represent pulses containing various parameters, also called features, such as frequency (F), pulse width, angle of arrival (AOA), and direction of arrival (DOA)-more details in Section 2.2. Over the years, many methods have been developed to deinterleave a RADAR signal, i.e., to separate the pulses from different emitters based on the PDW. Although one of the most reliable parameters for deinterleaving is the direction of arrival, only some methods are used because ESM systems do not often measure it.

At first, the systems were less sophisticated than now; simple and easily separable characteristics represented emitters as transmitted continuously on a single frequency (little noise, few adverse countermeasures). Early work from the 80s, often called "PRI-methods," mainly used temporal analysis to separate the pulses, using and exploiting the time of arrival (TOA). After filtering the frequency and the direction of arrival, the time of arrival is used to find the pulse repetition interval. The PRI corresponds to the emission pattern of the emitter and is usually very different, almost unique, between RADARs. The PRI is obtained by differentiating the successive pulses a RADAR emits in a signal to extract the emission pattern.

Several methods were developed among the "PRI-methods" and have been a foundation for many authors who have enriched the RADAR literature. For example, one have the cumulative difference histogram (CDIF) [Mar89] or the sequential difference histogram (SDIF) [MP92]. The authors develop methods by analyzing the first-order difference of the TOA sequence, which is the DTOA, to find peaks in the histogram and look for periodic PRIs according to a fixed threshold. Other methods have been proposed, such as using correlation function always from the TOAs [Sch74; NK00; Nel93], Kalman filter [MK94; CM98; Hoc98] or by representing the DTOA as a harmonic matrix [Ray98].

Note that many recently developed methods are still based on the analysis of TOA and propose improvements to these historical algorithms, such as the Multi-Level time-difference of arrival (TDOA) [Ge+19], dynamic sequence searching [Xi+17], autocorrelation functions [Cha+10; Cha+06], the Multi-level TDOA histogram [Ge+19] or many improved version [MP11; NK00; Mao+09] of the PRI transform aglorithm [Nis83].

Searching for the PRI pattern is straightforward for simple RADAR emitters because they transmit continuously with a single PRI. This research can quickly be complex due to various waveforms (multiple frequencies, agilities, sub-operation modes, etc.), disturbances, missing pulses, blanking, etc. This is all the more true as the quantity of data to be processed is greater, specifically in the case of methods based on autocorrelation function (high computational times).

A significant limitation of these methods is that they do not consider missing values when searching and extracting the PRI. There is a need to model this phenomenon-more information in Section 4.1.2. When intercepting a signal, it is possible to intercept only some pulses (detection threshold, noise, bad interception, interference, mixing pulses, etc.), complicating the search of the PRI. Therefore, the distribution obtained following the differentiation of successive temporal pulses of a RADAR, named DTOA or estimated PRI, from a signal does not necessarily correspond to the PRI of the RADAR. To summarize, the PRI refers to the emission characteristics of the RADAR coming from a database, while the values obtained following the temporal differentiation of successive pulses are the DTOA. Note that if a passive system intercepts all the pulses of a RADAR without noise in perfect conditions, the DTOA should correspond to the PRI.

New methods integrating other primary parameters, such as frequency or pulse width combined with deep learning models [Noo99; Liu+05] appeared quickly. In [AA07], the authors proposed clustering from the Direction of arrival and frequency with a Fuzzy ART before extracting the PRI pattern to identify the emitter. The Fuzzy architecture [KGG85; JM15; Wan+20] has been widely used in the last few years; unlike traditional neural networks that assign each input to a single class, Fuzzy ART allows input to belong to multiple classes simultaneously with fuzzy degrees of membership. However, these models are noise-sensitive, have long execution times, and require a lot of configuration.

In [LY19; LLH20], the authors present a method to simultaneously realize a signal's classification, denoising, and deinterleaving. As signals become increasingly complex, it is necessary to consider the features that can impact signal quality before applying deinterleaving techniques. However, their method can only handle a limited number of waveforms (constant or stagger pulse width and PRI). Few models developed can deinterleave a signal by considering various waveforms, and more specifically, the RADARs themselves.

The deinterleaving task is sometimes assimilated to image segmentation [Gas+20; Nuh+23]. In [Gas+20], The authors proposed a Neural Network-based clustering method for deinterleaving a signal by transforming the deinterleaving task into an image segmentation problem. The signal is encoded as an image using spectrograms based on the frequency and the time of arrival. Then, they apply a U-Net architecture to predict clusters instead of using clustering algorithms. Deep Learning models require considerable data and are difficult to parameterize. These methods are primarily built in a supervised framework, and their results strongly depend on the quality of the simulator or the training dataset used.

Deep learning models have been increasingly requested to manage the sophistication and emission complexity of emitters [Zho+18; ZWL22; Liu21; SS22; Yan+23; Kan+23]. Most of these methods are based on processing the PRI data [XSZ23]. When the signal is misestimated, it is possible to have missing pulses, which can damage the temporal patterns of the RADAR. Indeed, due to the complexity of the problem, supervised approaches, such as deep learning techniques, generally failed to provide correct deinterleaving performance.

However, methods based on clustering pulse features are based on models too simple to deal with the variety and complexity of RADAR signals [DLW22; SR21; SB23]. Indeed, these methods cannot identify RADAR emitters that emit on several different frequencies, pulse width, etc. This can be seen in the method developed by these authors [BE12]. The developed method is based on the Antenna Scan Pattern (ASP) to deinterleave a signal and on the Antenna Scan Type (ATP) for the classification. Although their results are encouraging, they emphasized that their methods had difficulty considering the frequency agilities of RADARs.

In recent years, new unsupervised or semi-supervised methods have started to emerge [ZLH17]. Supervised methods are applied to labeled data, and then a hybrid classification model built from several algorithms is used to improve classification accuracy and robustness. Data are truncated or partially observed, making pattern identification difficult. Gaussian Mixture Models (GMMs) are increasingly used to classify and cluster signals because latent variables could be introduced and considered missing data [RME18]. Comparisons have shown that deep learning models are not necessarily better than conventional and simpler models such as GMMs [GMW08] or by combining the Hidden Markov Model (HMM) and Expectation Maximization (EM) algorithm [LK98]. They have shown, for example, that when the number of transmitters present in the signal was below a certain threshold, GMMs performed better.

At the beginning of the chapter, it was explained that the deinterleaving step was often done simultaneously with the identification, but some dedicated methods exist. Identification is a delicate step because it is based on the deinterleaving results. Using various methods, features are extracted from a set of pulses for comparison to a reference database. Generally, the chosen characteristics are frequency and time of arrival.

For simple RADAR signals, time-frequency analysis is sufficient to identify the present emitter with great certainty [KO04; LK07; Zen+11; RR10].

Most methods are still based on Deep Learning models and consider a small number of RADAR classes [LK07; LY19; Gen+21; Din+18; NAA20]. These methods require a large dataset for the training step, and adding a new class requires the classifier to be retrained. In [KO04], the authors propose two methods to classify the emitters according to a database. The first is based on a linear discriminant analysis (LDA) to extract feature parameters from time of arrival and frequency for the classification. Then, the second method, named Karhunen-Loeve transformation (TKL), is based on applying a principal component analysis (PCA) to reduce the space. Their methods are based on discriminating only nine RADAR classes. In [WWZ17], the authors propose a complex architecture to distinguish only eight classes based on a supervised classifier. Finally, in [LY14], a Neural Network classifier based on entropy is proposed. The proposed architectures are increasingly complex but can only manage very few classes, also complicating the explainability of the results

Additionally, a challenge in processing RADAR data is acquiring real data, even more labeled real data, due to the sensitive nature of the application field. Algorithms and methodologies are often developed using small datasets or simulated data. Most previous methods are based on simulated data, and their result performance strongly relies on the simulator's accuracy. The use of simulated data and the complexity of the proposed methods complicate their reproducibility. It is complicated to be able to reproduce certain methods and evaluate them.

The emitters' profiles have become increasingly complex, enhancing the panorama of RADAR of new types with more varied patterns. New methods have been developed to compare the group characteristics to a known database, also allowing the detection of new emitters [Liu+05; AC21].

To conclude, through the various methods cited in this chapter, the literature constantly evolves to consider technological advances in electronic systems and integrate artificial intelligence algorithms. However, the new methods are almost exclusively based on features such as PRI without offering adequate pre-processing. Concerning emitters' identification (or specification), the methods are often developed in a supervised framework and can only be managed by around ten classes.

In this context, this research aims to introduce a novel RADAR Recognition Process using optimal transport distances. The Chapters 3 and 4 respectively present new simple methods to improve the deinterleaving in an unsupervised framework and identify emitters by overcoming all previously explained constraints.

3 - Deinterleaving RADAR pulses

This chapter introduces the initial phase of the RADAR Recognition Process, starting with deinterleaving a signal. In RADAR interception processing, the ESM operators identify the emitters' pulses in a signal to adapt their decision-making. ESM operators require relevant information to analyze an intercepted signal and extract the significant information to separate the pulses from the emitters in the signal without prior information; these conclusions may have dramatic consequences depending on the circumstances. The chapter is divided into two parts, introducing two unsupervised approaches accompanied by relevant illustrations based on a combination of clustering algorithms and optimal transport distances to deinterleave a signal.

The first approach, named HACOT, is based on a two-step strategy combining the unsupervised hierarchical density-based spatial clustering of applications with noise (HDBSCAN) [CMS13] with hierarchical agglomerative clustering (HAC) [Joh67; CPD20] integrating the optimal transport distances [Vil09; Bon+11]. The first step involves separating the pulses with a clustering algorithm based on two features: frequency and pulse. An essential constraint is that the pulses of two different emitters cannot belong to the same cluster. Then, as the emitters exhibit complex behaviors and can be represented by several clusters, a hierarchical clustering algorithm based on an optimal transport distance is applied to merge these clusters. The approach can handle moderately complex signals, but the modernization of technological equipment and noise can strongly corrupt the quality of the intercepted signals spreading the pulses; a 2-dimensional clustering cannot distinguish pulses from emitters with similar characteristics, specifically in some areas such as airports or ports.

The second contribution is a variant named IHACOT, incorporating a pre-grouping phase developed to handle more complex signals. The method is based on a 3-dimensional clustering based on three features: time of arrival, frequency, and pulse width to group pulses capable of better separate pulses. A first pre-grouping phase is implemented to group each characteristic of the same emitter with a classical hierarchical agglomerative clustering using Euclidean distance from frequency. Then, the previously developed hierarchical agglomeration clustering using optimal transport distances is applied to perform the final grouping.

As a dendrogram represents hierarchical clustering results, both approaches are based on a decisional model to detect a threshold to prune the dendrogram. Stopping fusions prematurely leads to grouping insufficient clusters and spreading pulses across multiple clusters. Conversely, delaying pruning can mix pulses from multiple emitters into a single cluster. Two approaches have been proposed in each section to deal with this challenge. The first approach estimates a threshold using three unsupervised metrics, analyzing the optimal transport distance and knowledge of the RADAR system environment. This method provides encouraging results when the emitters are sufficiently distinguishable as they are mainly based on analyzing the intra-cluster, the inter-cluster variances, and the optimal transport distances. Their effectiveness becomes significantly limited when the number of clusters analyzed is high or the emitters have similar characteristics. A new and more efficient version of this decision model using statistical tests was developed to overcome these limitations.

3.1 . Hierarchical agglomerative clustering combined with optimal transport distances

This section presents the first approach implemented, named HACOT, highlighted in Algorithm 8, to deinterleave a signal with an illustration.

As pointed out in Chapter 2, one of the challenges in signal deinterleaving is the dispersion through several groups of pulses from an emitter in the (f_n, pw_n) plane. A two-step strategy has been built, which requires few features and is proposed to overcome this problem. Firstly, the HDBSCAN algorithm is applied to group the pulses from the frequency and the pulse width. As several clusters can represent emitters, a grouping phase based on a hierarchical agglomerative clustering combined with optimal transport distances is applied to group clusters belonging to the same emitter.

Algorithm 8 Hierarchical agglomerative clustering using optimal transport distances to deinterleave emitter pulses - HACOT

Data : Set of pulses, X
Features :
- Frequency, f_n
- Pulse width, pw_n
- Level, g_n
- Time of arrival, toa_n

Parameters :
- Minimum number of points to form a cluster, $MinPts$
- Statistical test, $test$
- Confidence level, α
- Threshold, λ
- Non-parametric method, $method$

Procedure :

1. Pulses separation: apply HDBSCAN for all x_i of X from f_n and pw_n according to $MinPts$: C (Set of clusters)

2. Cluster aggregation: apply hierarchical agglomerative clustering based on optimal transport distances presented in Algorithm 9 considering α, $test$, $method$ and λ for all c_i of C from toa_n and g_n: Y (Set of aggregated clusters)

Result : Y, Deinterleaved sets of pulses.

3.1.1 . Data description

An intercepted signal in Figure 3.1 was simulated from the simulator presented in Section 2.4, gathering 8917 pulses from three emitters, with characteristics given in Table 3.1. The characteristics of the emitters, with multiple frequencies, are chosen to highlight the robustness of the developed method. The last line presents the different levels of the standard deviation of the features.

Emitter	Frequency (MHz)	Pulse Width (ns)	PRI (μs)	Pulses
0	956	2	5	614
1	982, 986 990, 995	48	2.71, 2.76, 2.8, 2.85, 2.91, 2.95, 3.02	5123
2	941, 943, 946	240	6	3183
3	2.12	3.33	5.9e4	

Table 3.1: Simulated emitters characteristics.

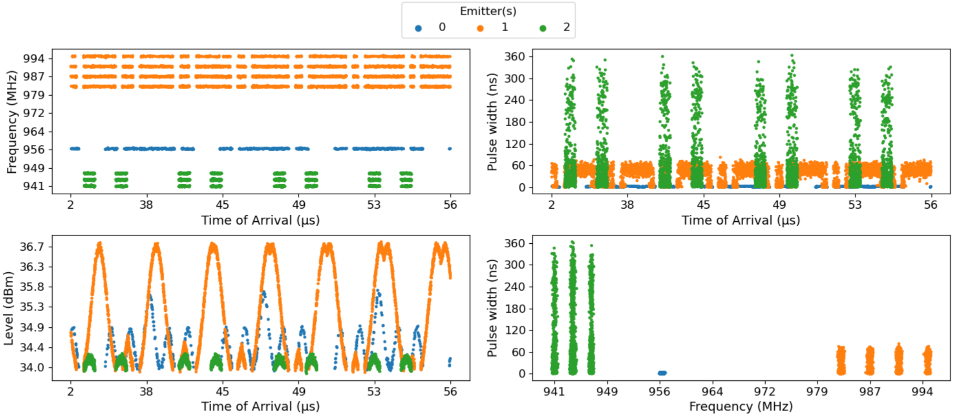


Figure 3.1: Example of a simulated signal gathering 8917 pulses of three emitters, identified by a color.

Frequency, level, and pulse width are plotted in function of time in panels of Figures 3.2, 3.3, and 3.4 respectively, and Figure 3.5 represents the (f_n, pw_n) plane. Each point represents a pulse, and the emitters are identified by a color (blue, green, and orange). An additional feature can be estimated from the intercepted signal: the difference of time of arrival $dtoa_n$, defined as the interval time between successive pulses belonging to the same emitter ($dtoa_n = toa_n - toa_{n-1}$). At this stage, the pulses are always mixed, meaning two successive pulses do not necessarily belong to the same emitter, making this feature unusable. Other features (waveforms, frequency modulation, etc.) will not be considered here. The approach considers only the four above-mentioned features because they are always available and relatively well-estimated. These representations highlight several challenging characteristics. Emitters transmit their pulses on different frequency bands, as underlined by emitter 1 in Figure 3.2, which is present on high frequencies and is characterized by four different frequencies. The signal also gathers pulses on two lower-frequency emitters, a mono-frequency emitter represented by emitter 0, while emitter 2 exhibits multiple frequencies.

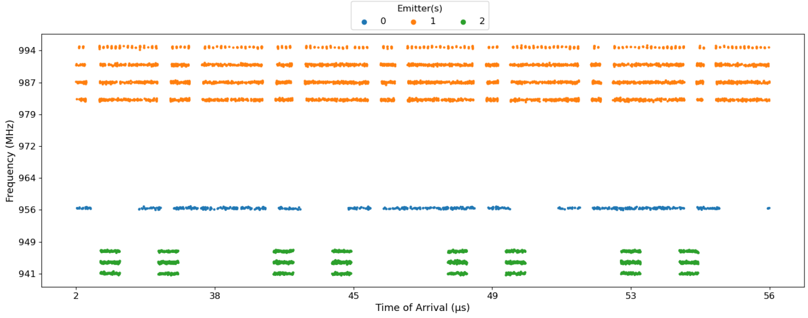


Figure 3.2: Simulated pulses of three emitters represented by a color in the frequency and time of arrival plane.

Pulse widths can be severely misestimated, as low-power pulses can be split during the segmentation stage, as shown in Figure 3.3. The pulse width spread is prominent for emitter 2, which exceeds 360 ns and 60 ns for emitter 1. Pulse-width spreading complicates emitter separability and can lead to misidentification.

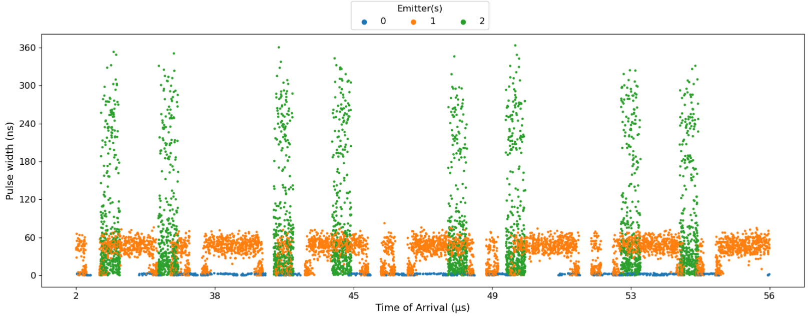


Figure 3.3: Simulated pulses of three emitters represented by a color in the pulse width and time of arrival plane.

Some emitters can be active simultaneously, leading to a superposition and a mixing of the lobes, as shown in Figure 3.4 plane. The side lobes of emitter 2 mix up with those of emitter 0 and 1, complicating their separability. The interception of pulses is not uniformly distributed across all emitters; the lobes of emitter 1 are distinctly visible (more than 5000 pulses), while the lobes of emitter 0 are discontinuous (around 600 pulses).

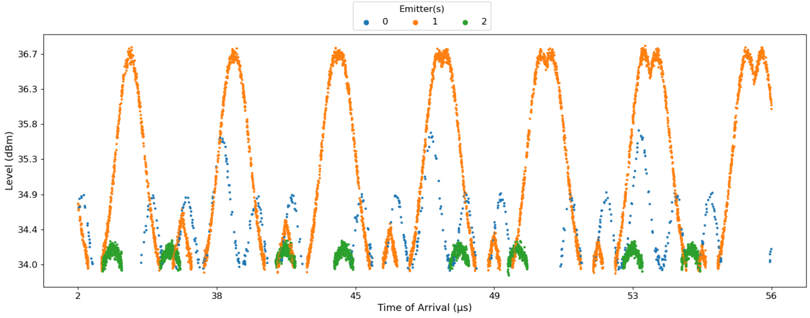


Figure 3.4: Simulated pulses of three emitters represented by a color in the level and time of arrival plane.

In Figure 3.5, Emitter 2 transmits on a single frequency and is represented by a single group of pulses. Conversely, emitters 0 and 1, characterized by several frequencies, have their pulses split across several groups, leading methods based only on clustering algorithms insufficient and inadequate for deinterleaving signals from this plane.

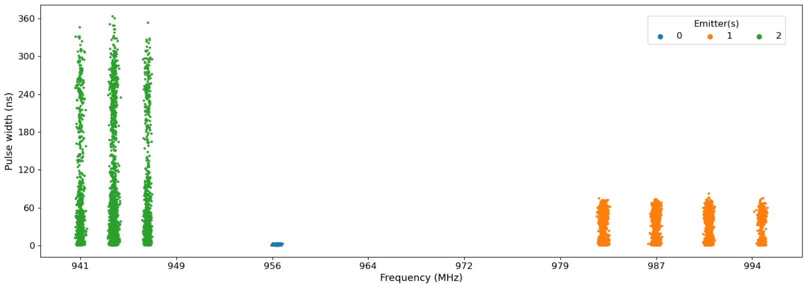


Figure 3.5: Simulated pulses of three emitters represented by a color in the frequency and pulse width plane.

In practice, the received pulses are unlabeled, and the deinterleaving step aims to separate the pulses and group them into different sets of pulses under the constraint of not merging the pulses belonging to several emitters into one set of pulses.

3.1.2 . Pulses separation with HDBSCAN in 2 dimensions from frequency and pulse width

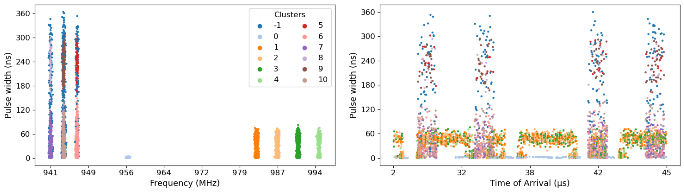
The first step of the method consists of applying a clustering algorithm to separate the pulses only from two highly discriminating and reliable characteristics: frequency and pulse width. Several algorithms, more detailed in Chapter 1.2, were tested to determine the most suitable data. Several clustering algorithm types are distinguished:

1. Partitioning algorithms: K-MEANS [HW79], or Hierarchical Agglomerative Clustering (HAC) [Joh67].
2. Probability density modeling algorithm: Gaussian Mixture Models (GMM) [MB88].
3. Density-based algorithms: Density-based spatial clustering of applications with noise (DBSCAN) [Est+96], Ordering points to identify the clustering structure (OPTICS) [Ank+99], or Hierarchical density-based spatial clustering of applications with noise (HDBSCAN) [CMS13].

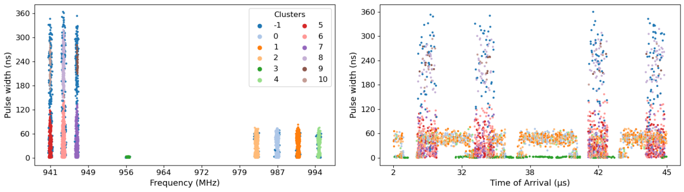
Partitioning algorithms and probability density modeling algorithms require assumptions about the number of clusters to be identified. Conventional methods, such as the Elbow method, unsupervised metrics, or data projection, have been used to fix the number of clusters to detect. The hierarchical approach is represented by a dendrogram aggregating each pulse step by step, requiring long time computing according to the signal size and a threshold to stop aggregations. Setting an appropriate threshold to prune the dendrogram presents a challenge, as it depends on the specific objectives or the data structure, particularly when dealing with noisy or poorly estimated signals. Mainly, these algorithms cannot detect clusters of different densities. Density-based algorithms do not necessarily require setting the number of clusters to detect and make no assumptions about their densities. Conventional clustering algorithms are tests whose results are highlighted in Figure 3.6. Each plot represents the results of a clustering algorithm and the color of detected clusters.

K-MEANS, GMM, and HAC results algorithms shown in Figures 3.6f, 3.6d and 3.6e provide coherent separation through 8 distinct clusters, avoiding mixing the pulses of several emitters in a cluster. These algorithms require parameterization of the number of clusters to detect and mostly make assumptions about the cluster's densities. The simulated signal presented in the developed approach is based on straightforward characteristics enabling pulse separation. Nevertheless, these algorithms are not able to detect outliers, and labeling all pulses can lead to poor pulse separation when the signal is noisy or contains outliers. This is all the more true when looking at the results in the (pw, toa) plane; these algorithms classified all points in the pulse with value is spread. The results of HDBSCAN and DBSCAN, shown in Figures 3.6a and 3.6b are very similar but detect outliers. The DBSCAN's outliers are concentrated in clusters characterized by high frequencies (over 979 MHz), while HDBSCAN perfectly groups these pulses across 4 clusters; the HDBSCAN's outliers are distributed in clusters below 949 MHz. OPTICS results seem better than those of HDBSCAN and DBSCAN in Figure 3.6c. Similar to the densities algorithms, the OPTICS results are very straightforward; it does not detect outliers due to the signal simplicity, leading to an excellent pulse separation. During the clustering phase, OPTICS presented several challenging drawbacks due to its parametrization setting and computation time precisely because the signal is extended.

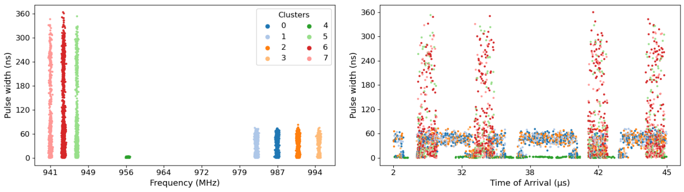
Moreover, in Figure 3.6, the algorithms OPTICS, K-MEANS, GMM, and HAC have grouped the pulses from the low-frequency emitter (below 949 MHz) into 3 clusters while HDBSCAN and DBSCAN separated these pulses into 6 clusters. This separation is preferable to avoid grouping pulses from several emitters in a single cluster.



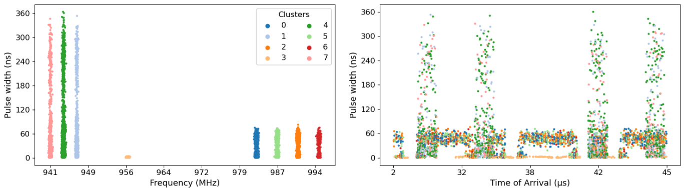
(a) HDBSCAN outputs - Outliers : 4.5%.



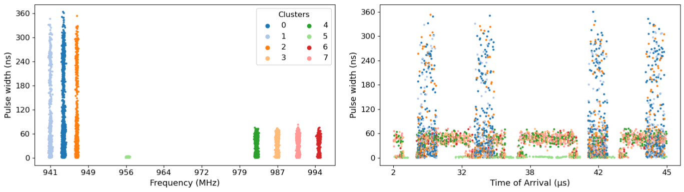
(b) DBSCAN outputs - Outliers : 4.4%.



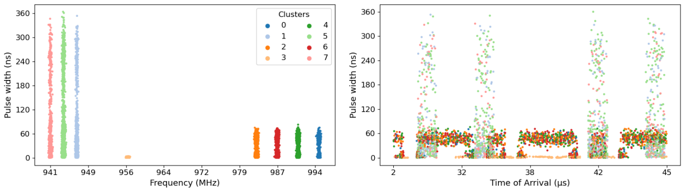
(c) OPTICS outputs - Outliers : 0%.



(d) GMM outputs - Outliers : 0%.



(e) HAC outputs - Outliers : 0%.



(f) KMEANS outputs - Outliers : 0%.

Figure 3.6: Results of clustering algorithms performed in (f_n, pw_n) plane, each color identifying a detected cluster.

All the algorithms manage to separate the pulses well across the groups; each group contains the pulses of a single emitter. The choice is based on the Hierarchical Density-Based Spatial Clustering of Applications with Noise algorithm (HDBSCAN)[CMS13], an unsupervised clustering algorithm based on a hierarchical version of the Density-based spatial clustering of applications with Noise (DBSCAN)[Est+96]. The choice of HDBSCAN is motivated by its capacity to identify clusters with different densities and shapes, the few parameters to optimize, its ability to detect the clusters without setting the number of clusters to be detected beforehand, and its capacity to deal with noisy signals.

The main idea is to group points that live in the same dense area from a vector $X = \{(f_1, pw_1), (f_2, pw_2), ..., (f_N, pw_N)\}$ grouping pulses thanks to the frequency and pulse width from a signal of N pulses. HDBSCAN results strongly depend on two hyperparameters, $MinPts$, and ϵ; see Section 1.2.6 for more details. HDBSCAN has been configured to overestimate the number of clusters returned so as not to mix the pulses of several emitters in the same cluster. Comparisons were made to find a tradeoff to fix this threshold; ϵ has been set to 0.01 and $MinPts$ to 20. For instance, some emitters can be characterized by few pulses or emitting very little over time. Comparisons were made to find a compromise to fix this threshold.

The physical dimensions of the frequency (MHz) and pulse width (ns) data are inconsistent. Renormalization is needed to calculate distances between features, specifically since the algorithms based on distance calculations are very sensitive to the data's dimensions, which can strongly impact their results. Multiples normalization methods are tested:

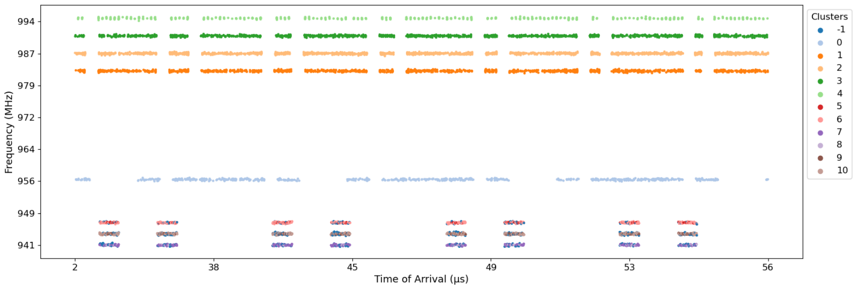
- Min-Max normalization: scaling each feature using a range
- Standard normalization: scaling each feature by removing the mean and scaling to unit variance.
- Max-Abs normalization: scaling each feature using its maximum absolute value
- Quantile normalization: scaling each feature using interquartile intervals
- Gaussian mixture models pre-clustering: estimation of the intra-cluster variances concerning the characteristics considered, based on GMM clustering with an overestimated number of clusters.

Clusters label	**Size**	**Clusters label**	**Size**
-1	402	5	100
0	614	6	623
1	1528	7	567
2	1548	8	86
3	1468	9	238
4	579	10	1164

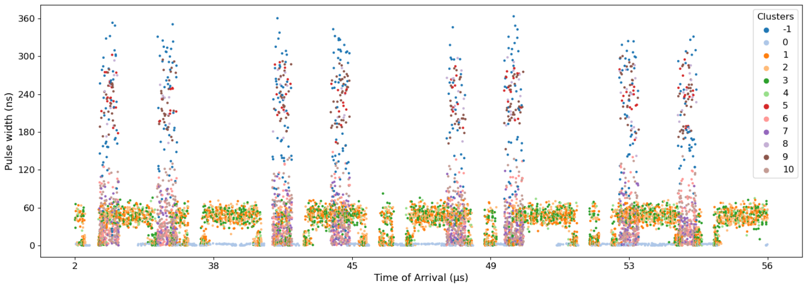
Table 3.2: Pulse distribution across HDBSCAN's resulting clusters.

Usually, the intercepted signal is noisy and $\backslash$ or includes outliers; scaling methods based on the average or absolute maximum values can distort the data distribution by including the noise or the outliers in the scaling. The pre-clustering method is time-consuming and necessitates a specific setting (e.g., the number of clusters to detect). The quantile method stands out due to its data distribution preservation, straightforward implementation, and robustness against outliers. The fixed nature of the quantile ranges is intentional, aimed at avoiding the exclusion of emitters with a low pulse count from the dataset. Specifically, employing the quantiles method as an interval for data normalization results in excluding Emitter 0 in Figure 3.5 due to its limited pulse count.

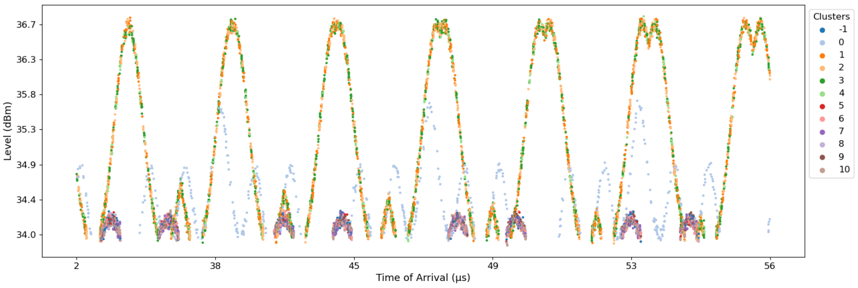
The results of the clustering performed by HDBSCAN in the (f_n, pw_n) plane are shown in Figure 3.7. As shown in Table 3.2, the clusters have heterogeneous sizes ranging from 86 to more than 1500 pulses and are represented by a color. Figure 3.7d shows the plane where the clustering was performed and that the clustering results are consistent. The pulses from emitter 0 transmitting around 956 MHz have been correctly grouped into a single cluster. Conversely, HDBSCAN splits the emitter pulses above 979 MHz and below 949 MHz into several clusters. Indeed, both emitters transmit on several frequencies. Additionally, low-power pulses of the lower frequency emitter are split at the segmentation stage, adding short pulse widths in the data. Figure 3.7c plane on the bottom left highlights the distribution of these clusters along the lobes.



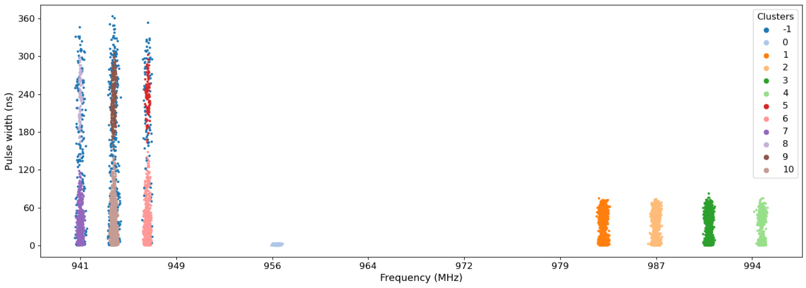
(a) Frequency x time of arrival plane.



(b) Pulse width x time of arrival plane.



(c) Level x time of arrival plane.



(d) Frequency x pulse width plane.

Figure 3.7: HDBSCAN outputs performed from frequency and pulse width plane. The algorithm identifies 11 clusters and an outliers class (-1), represented by a color.

3.1.3 . Cluster aggregation with hierarchical agglomerative clustering based on optimal transport distances from time of arrival and level

As shown in Figure 3.7d, an emitter can be split over several clusters. Therefore, the second step of the method is designed to group these clusters according to shared characteristics. In particular, it is assumed that the clusters of a given emitter are active simultaneously, as illustrated in Figure 3.7c where the pulses of an emitter are distributed over the lobes.

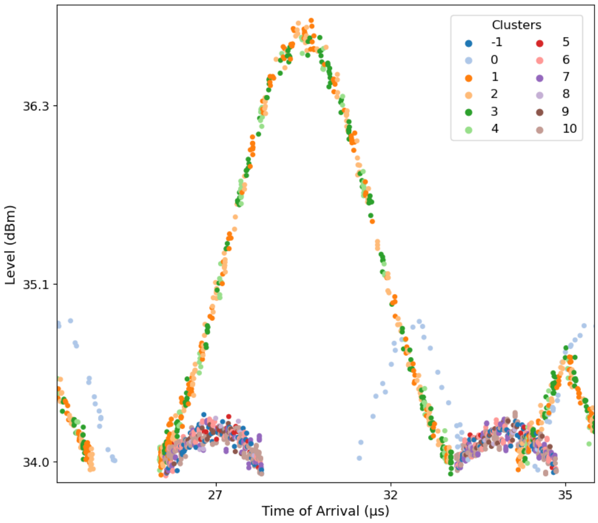


Figure 3.8: Zoom on HDBSCAN outputs in the (g_n, toa_n) plane.

From the HDBSCAN resulting clusters, a hierarchical agglomerative clustering based on the two features time of arrival and level was constructed [CPD20], as illustrated in the Algorithm 8 using the optimal transport distances [Vil09; Bon+11] to measure cluster similarity and dissimilarity. The following highlights the optimal transport distances suited to the context.

Algorithm 9 Cluster aggregation with hierarchical agglomerative clustering based on optimal transport distances

```
Data : Set of clusters, $C$
Features :
    - Level, $g_n$
    - Time of arrival, $toa_n$

Parameters :
    - Threshold, $\lambda$
    - Non-parametric method, $method$
    - Confidence level, $\alpha$
    - Statistical test, $test$

Procedure :
    1. Cluster significance analysis: separation of clusters according to
       $\lambda$: $D = \{c_i \in C \mid |c_i| \geq \lambda\}$ and $E = C \setminus D$
    2. Determination of a hierarchical structure aggregating the clusters
       using optimal transport distances from $D$: $structure$
       while $len(D) \neq 1$ do
           a) For all $d_i$ of $D$, representation by a measure from
              $toa_n$ weighted by $g_n$: $\tau_i$
           b) Computation of the distance for all $d_i$ of $D$
              using the optimal transport: $d(\tau_i, \tau_j)$
           c) Aggregation of the two closest clusters: $(i, j)^\star$
           d) Updating $D$
       end
    4. Dendrogram pruning from $D$ according to the decisional model
       presented in Algorithm 10 or Algorithm 11: $F$ (set of aggregated
       clusters)
    5. Dealing with excluded clusters $E$ with $method$:
        • Estimation of the probability density for all $f_i$ in $F$ from $toa_n$
          using $method$
        • Association of all $e_i$ of $E$ to $F$ by maximum likelihood estimation

Output : $Y$, aggregated clusters.
```

To define optimal transport distances between clusters, each cluster is represented by a probability measure describing its distribution from time of arrival weighted by level, as in Equation (1.1):

$$\tau = \frac{1}{Z}\sum_{p=1}^{P} g_p \delta_{toa_p}, \tag{3.1}$$

with P, the number of pulses in the cluster and $Z = \sum_{p=1}^{P} g_p$, the total energy in the cluster. In order to decrease the computational complexity of the method and avoid numerical problems, the probability measures used in practice are obtained from data histograms. The range of time of arrivals is partitioned in B intervals $[b_{i-1}, b_i]$. The measure τ in Equation (3.1) is replaced by

$$\bar{\tau} = \sum_{i=1}^{B} \bar{g}_i \delta_{c_i}. \tag{3.2}$$

Where $c_i = \frac{1}{2}[b_{i-1} + b_i]$, and $\bar{g}_i$ is the energy of the pulses in the interval, normalized so that $\sum_{i=1}^{B} \bar{g}_i = 1$. The number of the bins is fixed according to the Freedman–Diaconis rule [FD81]: $B = 2\frac{\mathrm{IQ}(N)}{\sqrt[3]{N}}$, with N the signal size and IQ the interquartile range of the signal. This method is less sensitive to outliers because it is based on the width of the IQ range, which is less influenced by outliers than the mean and standard deviation. In addition, it better adapts to non-Gaussian or skewed distributions by considering the real dispersion of the signal. This method adjusts for the sample size (other methods could be considered). Finally, the bin values are defined uniformly concerning the time of arrival values of the signal.

Figure 3.9 illustrates the use of optimal transport distances as an indicator to quantify similarity and dissimilarity between clusters similar to Figure 1.4. The Clusters 0, 3, and 4 distributions from Figure 3.7 are plotted in the (g_n, toa_n) plane. Clusters 3 and 4 distributions correspond to the same emitter; their distributions are highly similar and simultaneously active. Consequently, the cost of transporting the Cluster 4 distribution to the Cluster 3 distribution is low. Conversely, Clusters 4 and 0 distributions do not correspond to the same emitter. Here, pulses sent must be transported over a considerable distance, implying a higher total transport cost due to their different scan periods.

The optimal transport distance is used in an agglomerative clustering algorithm: the two clusters with the smallest optimal transport distance are

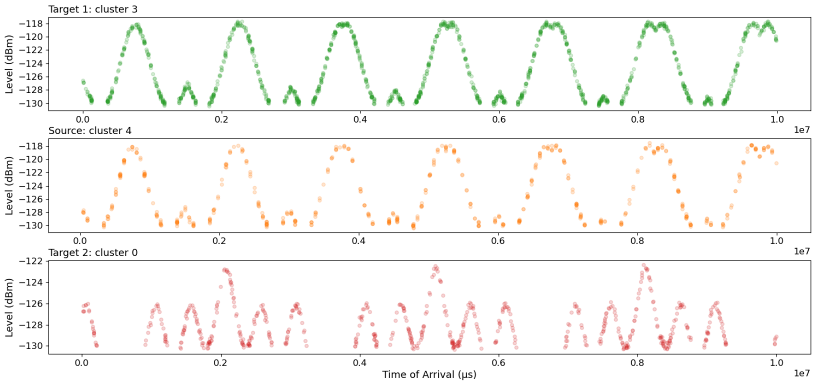


Figure 3.9: Distributions of Clusters 0, 3, and 4 in the (g_n, toa_n) plane.

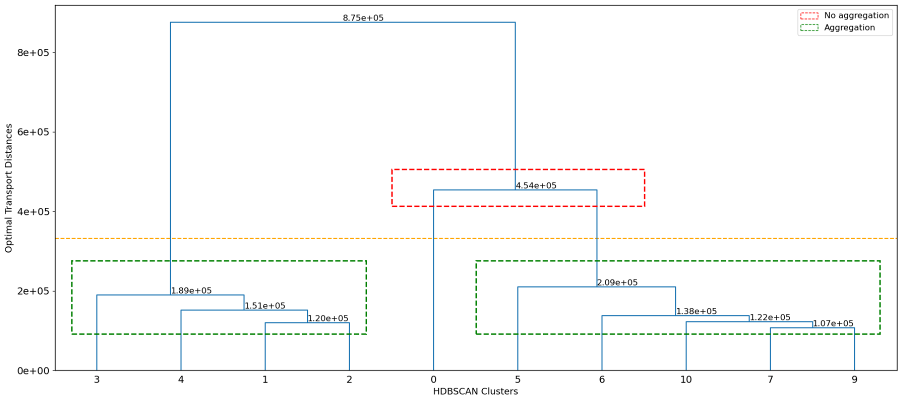
aggregated using Equations (1.3) and (1.4):

$$(i,j)^{\star} = \operatorname{argmin} d\left(\tau_i, \tau_j\right). \tag{3.3}$$

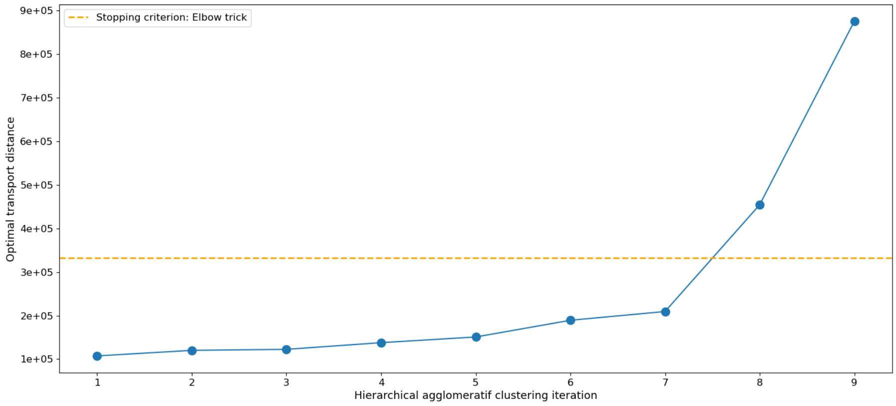
Note that the transport cost between clusters is symmetrical. After merging, the distances between merged clusters and other clusters are updated. Then, the process is repeated until all clusters are aggregated into one cluster. To obtain accurate estimates of the distance between clusters, each cluster must have sufficient pulses, as a probability distribution represents them. Several tests were conducted to establish a threshold; clusters having fewer pulses than this threshold are excluded from the hierarchical approach but will be processed separately using an alternative approach afterward. Here, Cluster 8, with 86 pulses, is set aside.

3.1.4 . Decisional Model based on hierarchical approach

The dendrogram in Figure 3.10a represents the aggregation at each step of the hierarchical agglomerative clustering combined with the optimal transport distances. The heights of the edges are the values of the optimal transport distances calculated at each aggregation, and these values are displayed. The final clustering is obtained by stopping the fusion of clusters at an appropriate step of the aggregation. Indeed, prematurely stopping the fusions may result in separating several emitters' pulses into multiple groups, and late fusion results in grouping pulses from different emitters in a single cluster.



(a) Dendrogram representing the aggregations at each iteration with the optimal transport distances values displayed, and the orange line highlights the elbow result.



(b) Optimal transport distances at each iteration; the orange line highlights the optimal distance gap.

Figure 3.10: Hierarchical agglomerative clustering combined with optimal transport distances results.

A first simple decisional model illustrated in Algorithm 10 based on searching the optimal transport distances gap and three metrics computed from the time of arrival and is proposed to stop fusions.

Algorithm 10 Decisional model for pruning the dendrogram based on unsupervised metrics and optimal transport distances

Data :

- Set of Clusters, D
- Hierarchical structure, $structure$

Features : Time of arrival, toa_n

Metrics :

- Silhouette Score, S_{SIL}
- Calinski-Harabsz Score, S_{CHL}
- Davies Bouldin Score, S_{DB}
- Gap Score, S_{GAP}

Procedure :

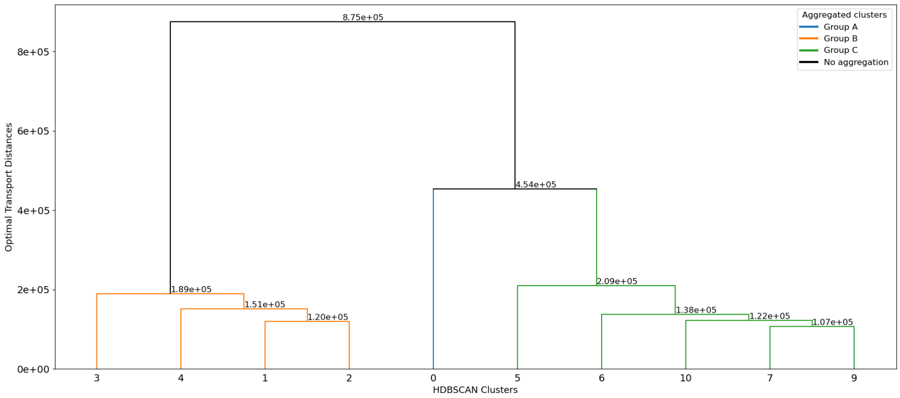
1. Estimation of $S^{\star}_{GAP}$ according the distances from $structure$ and adding in an empty vector: $\lambda = \{S^{\star}_{GAP}\}$.
2. Application of unsupervised metrics for all d_i of D according to the hierarchical structure: $S_{cores} = \{S_{SIL}, S_{CHL}, S_{DB}\}$ (Sets of vectors)
3. Estimation of the optimal thresholds per metric from S_{cores} and adding in $\lambda = \{S^{\star}_{GAP}, S^{\star}_{SIL}, S^{\star}_{CHL}, S^{\star}_{DB}\}$ (Set of values)
4. Determination of the optimal threshold based on majority rule from λ: $\lambda^{\star}$
5. Pruning $structure$ with $\lambda^{\star}$ on D: Y.

Output : Y: Set of aggregated clusters

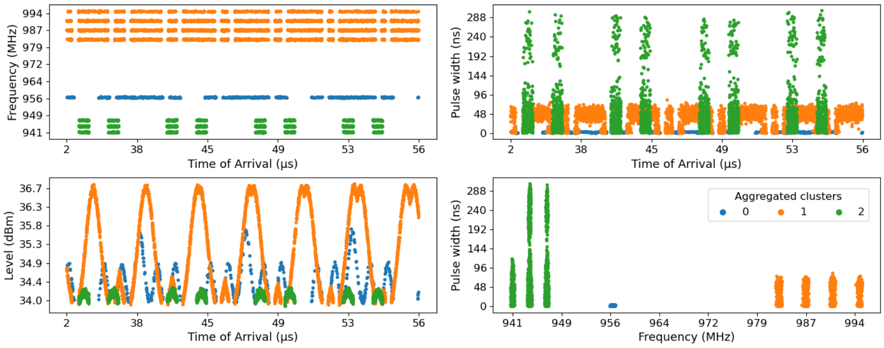
First of all, the optimal transport distances are analyzed to identify the break-point with the elbow rule; this method consists of plotting the optimal transport distances at each iteration and identifying a break-point in the evolution of the distance as shown in Figure 3.10b. A distortion appears between the 7th and 8th iterations, symbolized by the orange line; aggregating the Clusters at the 8th iteration requires many displacements to match the two distributions, indicating the two clusters do not correspond to the same emitter. This result is analyzed by considering the dendrogram in Figure 3.10a to identify the number of emitter present; the aggregations of clusters 5, 6, 10, 7, and 9, symbolized by the green box in Figure 3.10a, closely related optimal transport distances, implying the points require minimal displacements since these clusters correspond to the same emitter. Conversely, the optimal transport distance to perform the groupings of clusters 5, 6, 10, 7, 9, and 0, illustrated by the red box, is higher, indicating these clusters correspond to 2 different emitters and require more significant displacement. Finally, $S^{*}_{GAP} = 3$.

Then, the three unsupervised metrics are computed for each aggregation: the Silhouette score [Rou87], the Davies-Bouldin score [DB79], and the Calinski-Harabasz Score [CH74]. The metrics are applied at the time of arrival at each step of the aggregation. The metrics' optimal thresholds are obtained by maximizing the Silhouette and Calinski-Harabasz scores and minimizing the Davies-Bouldin score, $\{S^{\star}_{SIL} = 2, S^{\star}_{CHL} = 3, S^{\star}_{DB} = 2\}$.

By applying the Algorithm 10, the decisional model indicates the presence of three different emitters, and the final groupings obtained from the dendrogram are displayed in Figure 3.11b. The pulses of the emitters transmitting below 949 and above 979 MHz have been grouped together, as shown in Figure 3.11a; each color identifies the aggregations performed by the algorithm. However, the results of the decision model could be inconclusive and not handle a large signal complexity. The results of unsupervised metrics strongly depend on the distance chosen to assess the similarity between signal pulses (sensitivity to outliers, noise, etc.). They may fail to detect irregular cluster shapes or varying densities. The Silhouette score requires longer computation times, making its application insufficient for ESM operators to make real-time decisions. When the emitters are simultaneously active and have similar characteristics, the optimal transport distances obtained between their clusters can be close, complicating the break-point detection. Finally, determining the optimal threshold for pruning the dendrogram is an essential limit; in the developed decisional model, fusions were stopped when an estimated threshold between clusters was reached, meaning no more aggregations that passed this threshold were considered. Some signals require aggregating branches and not others.



(a) Dendrogram with the color identifying an aggregated cluster.



(b) Aggregated clusters get from the decisional model.

Figure 3.11: Hierarchical agglomerative clustering combined with optimal transport distances results obtained from the decisional model for pruning the dendrogram based on unsupervised
metrics and optimal transport distances.

To overcome these limitations, a new efficient model, highlighted in Algorithm 11, is developed, using a statistical test to compare the cluster's distribution based on the time of arrival. Multiple tests are evaluated as the Kolmogorov-Smirnov test [Mas51] and the Epps-Singleton test [ES86]. The two tests aim to compare the distributions of two samples to determine if they come from the same dataset but with two close and similar methods; the Kolmogorov-Smirnov test is based on analyzing distribution functions, while the Epps-Singleton test is on the underlying empirical characteristic functions. The tests are applied at each iteration of the hierarchical agglomerative clustering as the unsupervised metrics to determine if the two clusters must be aggregated, and their values are presented in the left plots on Figures 3.14.

The p-values are calculated at each step of the HACOT and compared to a confidence level fixed according to the test's value for each iteration. An aggregation will occur when the test's p-value is significant between two clusters. No aggregation occurs when no significant p-value is obtained. If the result is significant, the next iteration is considered until reaching the top of the tree or an insignificant result. When a non-significant result is reached, all subsequent aggregations involving these two clusters are no longer considered. The process selects the next iteration, and the algorithm continues.

Algorithm 11 Improved decisional model for pruning the dendrogram based on statistical test.

```
Data :
    - Set of clusters, $D$
    - Hierarchical structure, $structure$
Features : Time of arrival, $toa_n$
Parameters :
    - Statistical test, $test$
    - Confidence level, $\alpha$
Initialization : Create a list with all aggregations labels from $structure$
                 and an empty list: $N$ and $L$
Procedure :
while $len(N) \neq 0$ do
    Selection of the first two aggregated clusters according to $structure$: $\{i, j\}$ in $N$
    Representation by a measure from $toa_n$: $\theta_i$ and $\theta_j$
    Computation of the statistical test $test$ between $\theta_i$ and $\theta_j$ and get $p$-value
    if $p$-value $< \alpha$ then
        Remove all aggregations from $N$ using these clusters
    else
        Add $\{i, j\}$ in $L$ and del $\{i, j\}$ from $N$
    end
    Aggregate $l_i$ in $L$ according $structure$: $Y$
end
Output : $F$: Set of aggregated clusters
```

Standard values such as 1% or 5% are commonly used to compare the p-value and determine the test's significance. RADAR signals differ, leading to an impossibility and a nonsense in setting a singular threshold for all signals; a new, more efficient methodology is introduced to set a personalized threshold for each signal and test. The p-values obtained at each iteration of the HACOT are sorted to determine a break-point to set the evaluation threshold of the test. Figures 3.13 plot the sorted p-values of the Kolmogorov-Smirnov and the Epps-Singleton test. In this example, the confidence level is fixed to $\alpha_{ks} = 0.0104$ for the Kolmogorov-Smirnov test and $\alpha_{epps} = 0.0198$ for the Epps-Singleton test. The orange line symbolizes the thresholds. The break-point is readily observable by analyzing the values of the Kolmogorov-Smirnov test, while the difference is less pronounced for the Epps-singleton test.

The values of the p-values do not decrease with the value of the iteration as in Figure 3.10b. The p-values of the last iterations are expected to be equal or close to 0 because the clusters merged are expected to not belong to the same RADAR as shown in Figure 3.12.

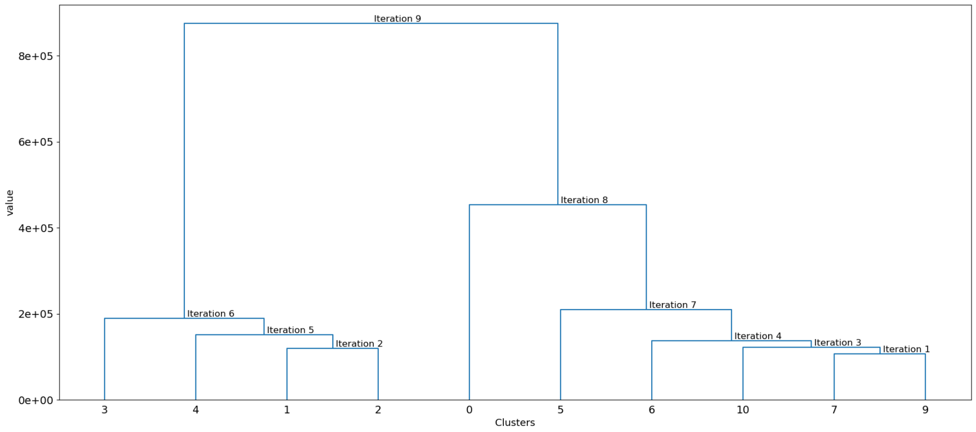
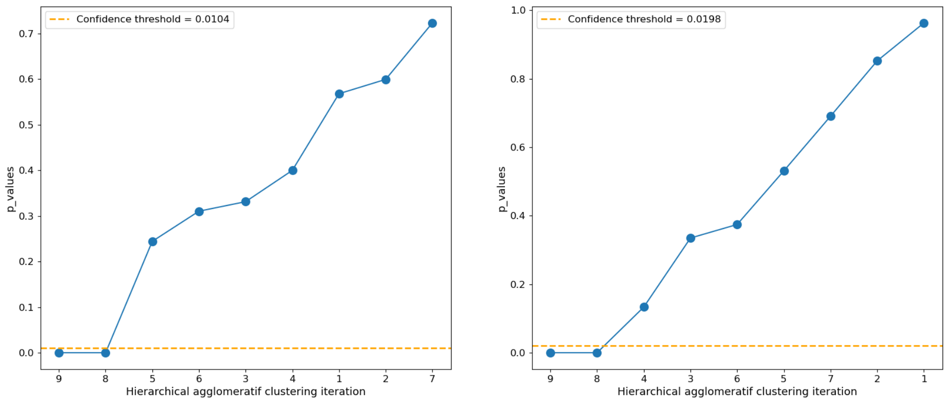


Figure 3.12: Dendrogram showing iteration numbers of the hierarchical agglomerative clustering using optimal transport distances.

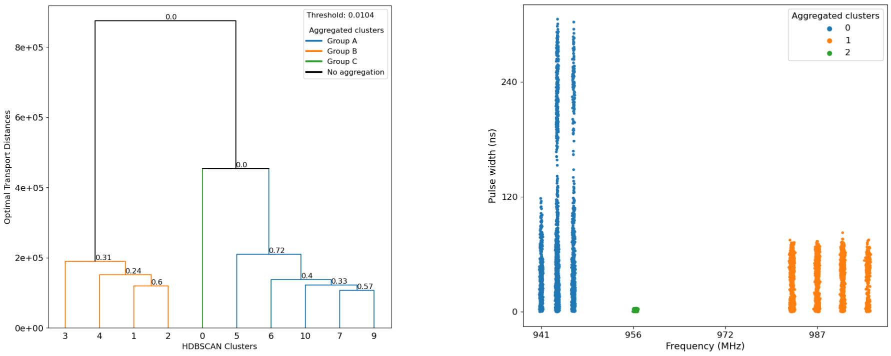
After fixing a threshold per test, the evaluation process starts following the results of the dendrogram in Figure 3.11a by selecting the two aggregated clusters 7 and 9 and representing them with a measure from the time of arrival. The associated result for the Kolmogorov-Smirnov is 0.57, which is higher than our $\alpha_{ks} = 0.0104$ threshold, meaning Clusters 7 and 9 distributions correspond to the same emitters; the two clusters will be aggregated. The process is repeated until the top of the tree is reached or a non-significant result is returned. The p-values between Clusters 0 and 5, 6, 10, 7, and 9 are 0, which means the clusters should not correspond to the same emitter. The process stops, and the next iteration is evaluated. The decisional model identified three aggregated groups. A similar process is applied using the Epps-Singleton test and presented in Figure 3.14b. The right plots in Figure 3.14 display final groupings are identical to those obtained using the Kolmogorov-Smirnov test.



(a) Kolmogorov-Smirnov test.

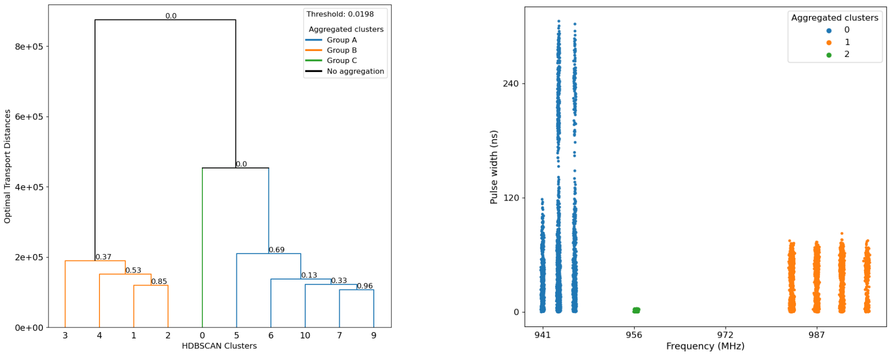
(b) Epps-Singleton test.

Figure 3.13: p-values sorted according to the test, and the orange line highlights the estimated confidence level.



Dendrogram. Aggregated clusters.

(a) Kolmogorov-Smirnov test.



Dendrogram. Aggregated clusters.

(b) Epps-Singleton test.

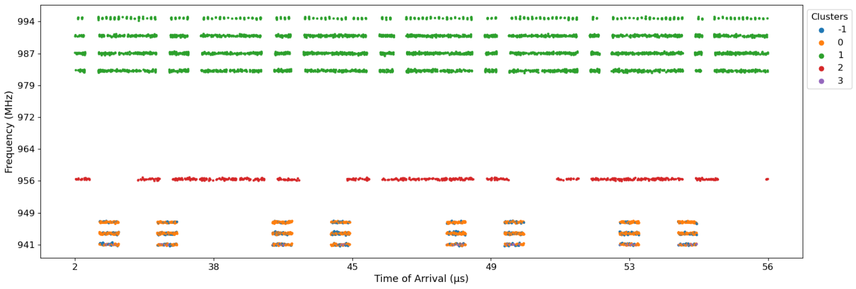
Figure 3.14: Dendrograms with the associated p-values of the test and the representation of the aggregated clusters in the (f_n, pw_n) planes.

3.1.5 . Dealing with excluded clusters

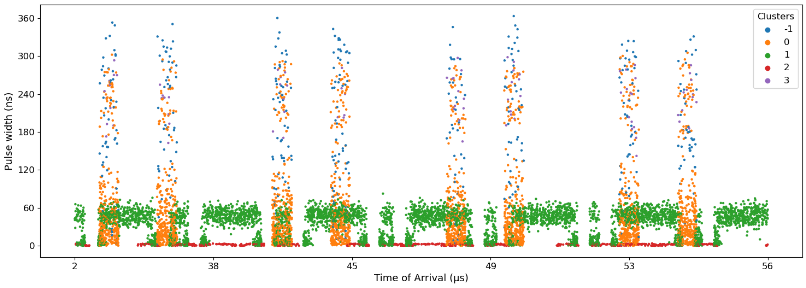
As previously mentioned, Cluster 3 was set aside from the analysis due to its insufficient number of pulses. Clusters excluded from hierarchical clustering are now associated with an extensive cluster obtained after the grouping phase. The process is done in two steps:

1. Estimation of the probability density of time of arrival of the extensive clusters using a kernel density estimator [Ros56; Par62] with a Gaussian kernel.
2. Association of an excluded cluster to an extensive cluster by maximum likelihood estimation.

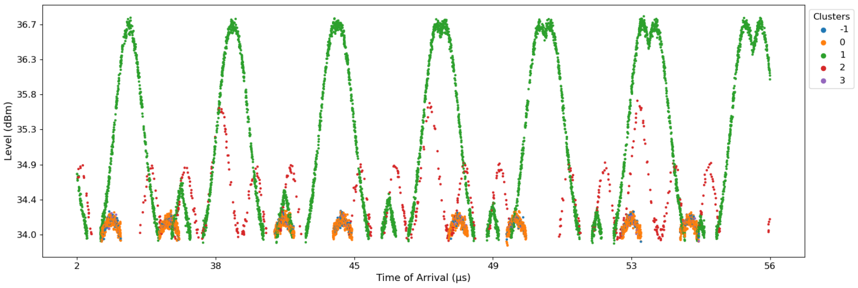
Figure 3.15 plots the pulses of the aggregated clusters and the excluded clusters. The emitter's pulses around 956 Mhz and above 979 Mhz in Figure 3.15d have been perfectly grouped into two sets of pulses. Conversely, the emitter below 949 MHz has these pulses split into two clusters.



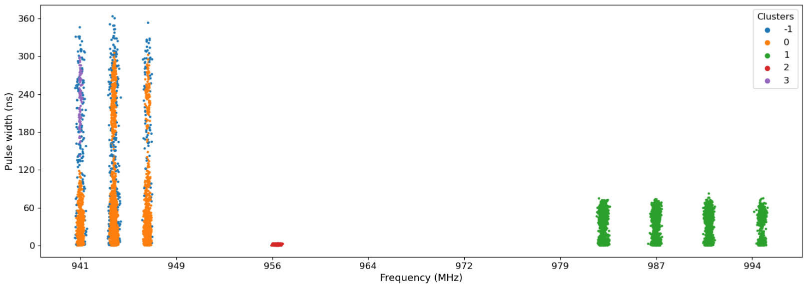
(a) Frequency x time of arrival plane.



(b) Pulse width x time of arrival plane.



(c) Level x time of arrival plane.



(d) Frequency x pulse width plane.

Figure 3.15: Aggregated cluster obtained following the dendrogram from the hierarchical agglomerative clustering using optimal transport distances with excluded clusters. Each color identified an aggregated cluster with an outliers class (-1).

A non-parametric model is used to estimate the distribution parameters and their density function. Among these methods, the most commonly used are that of histograms [Sco79; Par62], K-nearest-neighbors [MR79; FH73], Kernel Density Estimation [STT80] or Neural Networks [MA98]. The kernel density estimation (KDE) is selected because it allows us to build a density from each point and, therefore, to better consider the distribution's behavior. For a sample of points $(x_0, x_1, ..., x_N)$ belonging to a cluster distribution f, the kernel density estimator is written as:

$$\hat{f}_i(X) = \frac{1}{Nh} \sum_{i=0}^{N} K\left(\frac{x - x_i}{h}\right), \tag{3.4}$$

with N the number of pulses, K the kernel, and h the smoothing parameter (bandwidth). The KDE mainly depends on two parameters: the choice of the kernel and the bandwidth. For a Gaussian kernel, by assuming that a Gaussian distribution can represent the time of arrival, the optimal bandwidth can be obtained from the Scott method [Sco15] and is given by

$$\begin{aligned} h_i &= \left(\frac{4}{3n}\right)^{1/5} \hat{\sigma_i} \\ &\approx 1.06 \hat{\sigma}_i n^{-1/5}, \end{aligned} \tag{3.5}$$

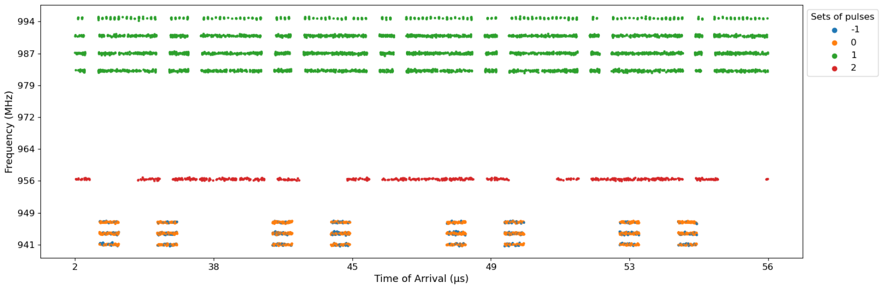
with $\hat{\sigma}$ the standard deviation of the cluster. As the distribution of the TOA for a given emitter is multimodal, the assumption is made that the distribution of the TOA is a mixture of Gaussian. Temporal clustering with HDBSCAN for each extensive cluster is applied to identify each appearance and estimate the variance of each sub-cluster. The parameter $\hat{\sigma}$ in the bandwidth estimation is then obtained by averaging the variance of each sub-cluster. Finally, for a given excluded cluster, the likelihood of belonging to an extensive cluster i is given by

$$L_i = \prod_{n=1}^{N} \hat{f}_i(toa_n), \tag{3.6}$$

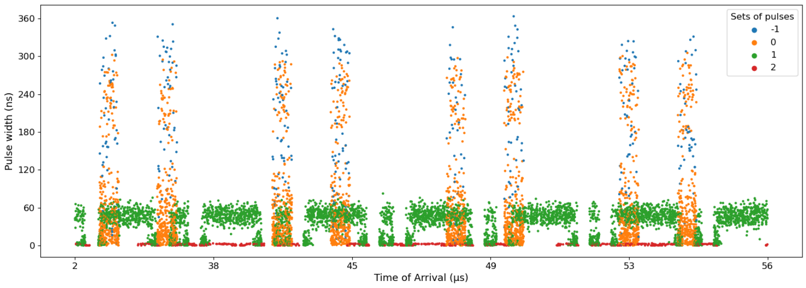
with toa_n the time of arrival. The excluded cluster is assigned to the extensive cluster, maximizing the likelihood:

$$i^\star = \operatorname{argmax} L_i, \tag{3.7}$$

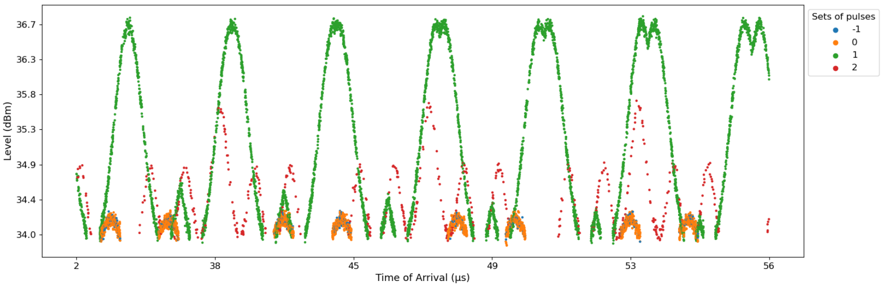
with the likelihood of the considered cluster belonging to the extensive cluster i. Cluster 3 will finally be associated with Cluster 2.



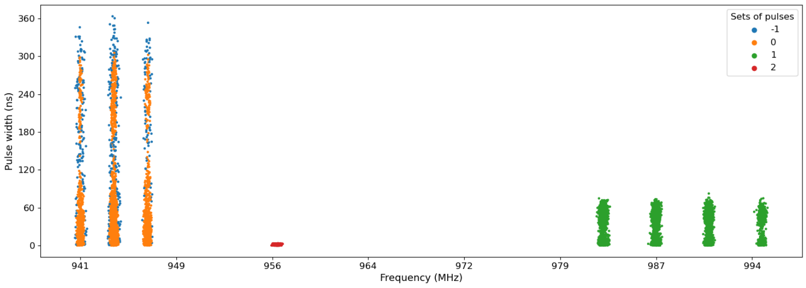
(a) Frequency x time of arrival plane.



(b) Pulse width x time of arrival plane.



(c) Level x time of arrival plane.



(d) Frequency x pulse width plane.

Figure 3.16: Final grouping after applied hierarchical agglomerative clustering combined with optimal transport distances and kernel density estimation on excluded clusters. Each color identified a set of pulses with an outliers class (-1).

Finally, Figure 3.16 displays the cluster's agglomeration. The hierarchical agglomerative clustering combined with the optimal transport distances and the processing of the excluded clusters correctly separates the pulses in the signal, groups them into three sets of pulses, and can handle clusters with few pulses. The values of homogeneity and completeness are equal to 1, meaning that the pulses are correctly distributed in the different sets of pulses, and there is no mixing. The outlier rate indicates that 4.5% of the pulses are lost during the deinterleaving process; HDBSCAN excluded these pulses in step 1.

3.1.6 . Computational complexity

The computational complexity of the first method can be estimated in function of the number N of pulses, the number of clusters C identified by HDBSCAN, and the number of bins B. The complexity of the first stage, dominated by HDBSCAN, is $O(N^2)$ [Cam+15]. In the second stage, C^2 distances are computed at the beginning of the hierarchical clustering, each costing $O(B)$ using the formulation of the Wasserstein distance as a distance between quantile functions. The number of updated distances during the clustering is also in $O(C^2)$.

The actual clustering, with complexity in $O(C^3)$, can be neglected under the reasonable assumption that the number of bins B is larger than the number of clusters. The ($C-1$) Komolgorov-Smirnov tests and the treatment of the excluded clusters are negligible compared to the initial distance computation.

The total complexity of the algorithm is thus $O(N^2 + C^2 B)$. This complexity can be rewritten in functions of parameters related to the characteristics of the emitters. With T the length of the signal, K the number of emitters, $\bar{R}$ the mean rate of pulses, $\bar{C}$ the mean number of clusters per emitter (i.e., an evaluation of the complexity of the emitter), the total complexity is $O(K^2(T^2\bar{R}^2 + \bar{C}^2 B))$.

3.2 . Improved Hierarchical agglomerative clustering combined with optimal transport distances

In some cases, pulses associated with two different emitters can be grouped in the same cluster by HDBSCAN. This can arise when the emitter has similar characteristics and/or inaccurate estimations of the parameters of the pulses are performed. The 2-dimensional clustering done by HDBSCAN will return close clusters in the (f_n, pw_n) plane. This proximity is problematic when the signal includes noise; the pulses appear as one large burst of pulses; HDBSCAN cannot separate the pulses from these emitters and groups them into a single cluster. This section introduces a variant of the method to alleviate this issue, detailed in Methodology 12; it differs from the previous method by using the time of arrival in the first clustering steps and adding a pre-grouping step before applying the Algorithm 8 previously developed in Section 3.1.

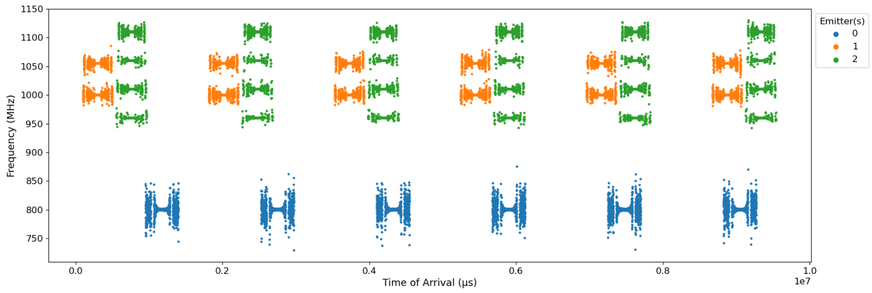
3.2.1 . Data description

An intercepted signal in Figure 3.17 was simulated from the simulator presented in Section 2.4, gathering 22760 pulses from three emitters, with characteristics given in Table 3.3. The characteristics of the emitters, with multiple frequencies, are chosen to highlight the robustness of the developed method. The last line presents the different levels of the standard deviation of the features.

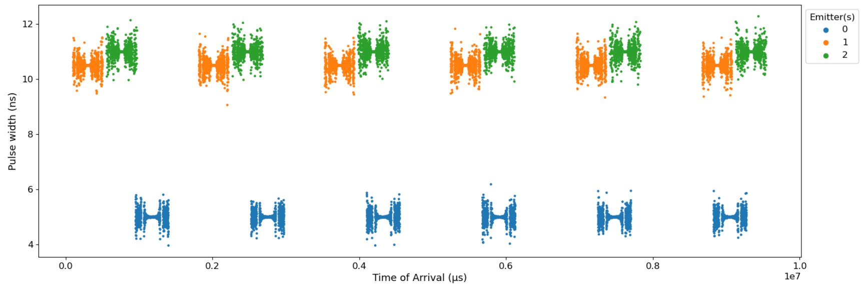
Emitter	**Frequency (MHz)**	**Pulse Width (ns)**	**PRI (μs)**	**Pulses**
0	800	5	300	7462
1	1000, 1055	10.5	200	7892
2	960, 1010 1060, 1110	11	200	7406
Std	121.7	2.7	3.2e4	

Table 3.3: Simulated emitters Characteristics.

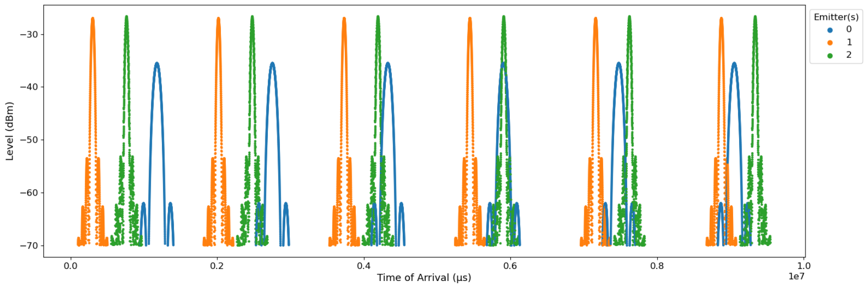
After the signal acquisition, the pulses are described by the same features as the previous Section 3.1. Each point represents a pulse, and each color corresponds to one emitter. The features are plotted as a function of time on the three plots, and the last plot represents their frequencies according to their pulse widths.



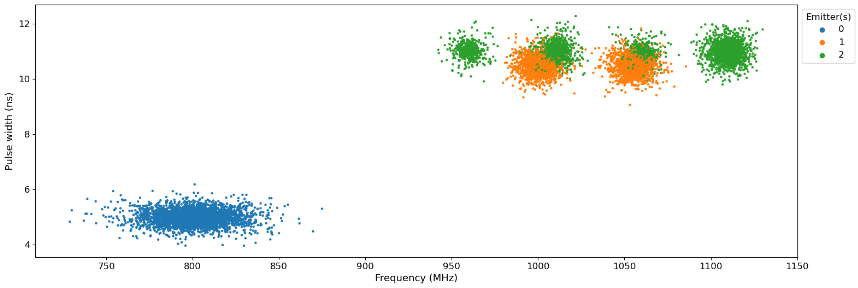
(a) Frequency x time of arrival plane.



(b) Pulse width x time of arrival plane.



(c) Level x time of arrival plane.



(d) Frequency x pulse width plane.

Figure 3.17: Example of a simulated signal gathering 22760 pulses of three emitters, represented by a color.

In Figure 3.17a, the Emitter 0 (blue) is easily identifiable and characterized by low frequencies (around 800 MHz). In comparison, the pulses of the two other emitters share the same frequency bands apart from 900 MHz. The pulse width spread and the simultaneity of emission of the two emitters emitting above 900 MHz, presented respectively in Figures 3.17b and 3.17c planes, are not sufficient to differentiate them. The spreading of the pulses is caused by the presence of noise in the signal; this result is confirmed by looking at the pulses spread in Figure 3.17d with the pulses overlapping, leading to an inseparability of these two emitters.

Algorithm 12 Improved hierarchical agglomerative clustering using optimal transport distances to deinterleave emitter pulses - IHACOT

Data : Set of pulses, X

Features :

- Frequency, f_n
- Pulse width, pw_n
- Level, g_n
- Time of arrival, toa_n

Parameters :

- Minimum number of points to form a cluster, $MinPts$
- Distance, $dist$
- Statistical test, $test$
- Confidence level, α
- Threshold, λ
- Non-parametric method, $method$

Procedure :

1. Pulses separation: apply HDBSCAN for all x_i of X from toa_n, f_n, and pw_n: C (Set of clusters)

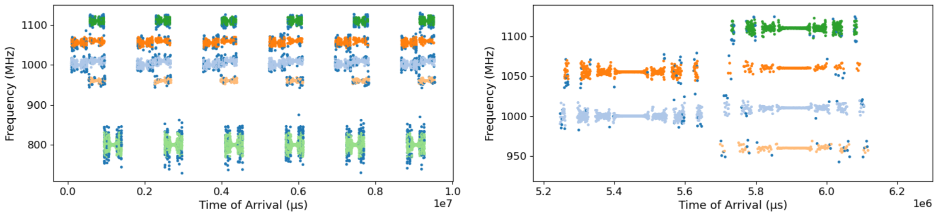
2. Pre-clusters aggregation: apply hierarchical agglomerative clustering based on $dist$ presented in Algorithm 13 for all c_i of C: P (Set of pre-aggregated clusters)

3. Cluster aggregation: apply hierarchical agglomerative clustering based on optimal transport distances presented in Algorithm 9 considering α, $test$, $method$ and λ for all p_i of P from toa_n and g_n: Y (Set of aggregated clusters)

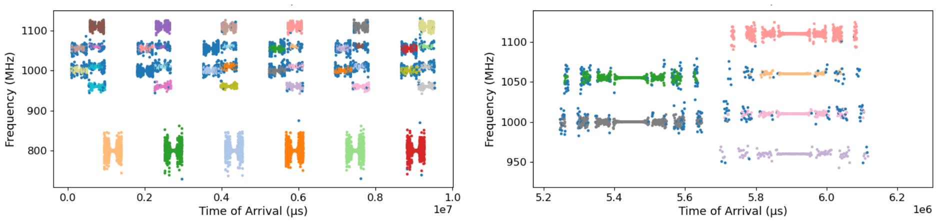
Result : Y, Deinterleaving sets of pulses.

3.2.2 . Pulses separation with HDBSCAN in 3 dimensions from time of arrival, frequency, and pulse width

When noisy signals, measurement errors can appear on the estimated frequency and pulse width, leading to a spreading of the pulses. The 2-dimensional clustering returns close clusters as shown in Figure 3.18a, meaning the pulses can not be adequately separated. Therefore, it becomes imperative to add additional features to the clustering.



(a) HDBSCAN results performed from frequency and pulse width.



(b) HDBSCAN results performed from the time of arrival, frequency, and pulse width.

Figure 3.18: Comparison of HDBSCAN clustering performed in 2 and 3 dimensions. The right plots show a zoom.

A three-dimensional vector $X = \{(toa_1, f_1, pw_1), ..., (toa_N, f_N, pw_N)\}$ grouping time of arrival, frequency, and pulse width pulses from a signal of N pulses is passed as input in HDBSCAN to be clustered. Incorporating the time of arrival in the clustering improves emitters' pulse discrimination, as shown in Figure 3.18b. The clustering results obtained in 2 dimensions from (f_n, pw_n) plane are displayed on the left plot; HDBSCAN identifies 5 clusters with an outliers class. The 2-dimensional clustering mixed the pulses of the two emitters at 1000 and 1060 MHz. Conversely, adding a third feature improves the separation, as highlighted by the right plot; the algorithm identifies 42 clusters with a class of outliers (-1).

3.2.3 . Pre-clusters aggregation with hierarchical agglomerative clustering based on Euclidean distance from frequency and pulse width

A first grouping phase in 3-step, highlighted in Algorithm 13, is implemented before applying the Algorithm 8 previously developed. First, the clusters are spaced to avoid their overlapping by computing their averages from their frequencies ($\overline{f_n}$) and pulse width ($\overline{pw_n}$) and representing them in the ($\overline{f_n}, \overline{pw_n}$) plane. From this new plane, a hierarchical agglomerative clustering using Euclidean distances is applied to group the clusters by stopping the aggregations by applying the decision model developed in the Algorithm 11 from the frequency.

Algorithm 13 Pre-clusters aggregation with hierarchical agglomerative clustering based on Euclidean distance

Data : Set of clusters, C
Features :

- Frequency, f_n
- Pulse width, pw_n

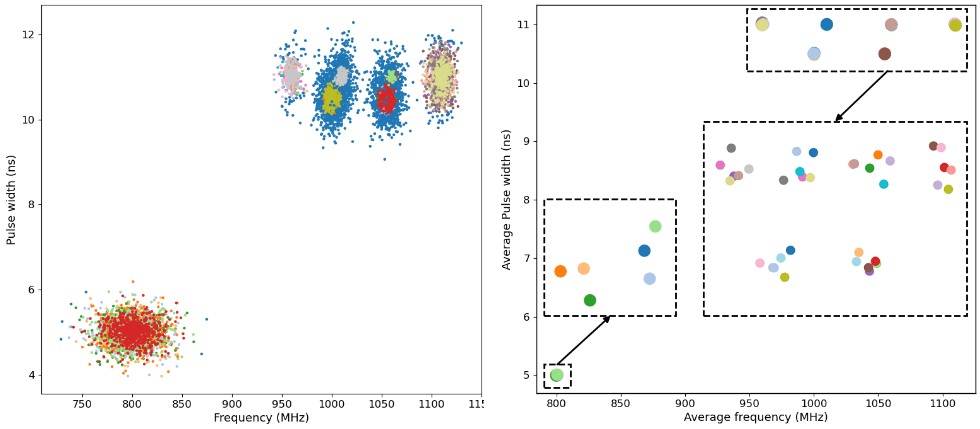
Parameters :

- Distance, $dist$
- Statistical test, $test$
- Confidence level, α

Procedure :

1. Estimation of $\overline{f_n}$ and $\overline{pw_n}$ for all c_i of C: $\overline{C}$ (Set of average values per clusters)
2. Hierarchical agglomerative clustering using $dist$ for all $\overline{c}_i$ of $\overline{C}$ using $\overline{f_n}$ and $\overline{pw_n}$
3. Pruning the dendrogram by applying the improved decisional model developed in Algorithm 11 from f_n considering $test$ and α: P (Set of pre-aggregated clusters)

Output : P: Set of pre-aggregated clusters.



(a) Clusters representation in the (f_n, pw_n) plane.

(b) Cluster averages' representation in the (f_n, pw_n) plane.

Figure 3.19: HDBSCAN clustering results performed in 3 dimensions plotted in the (f_n, pw_n) plane on the left and cluster averages in the (f_n, pw_n) plane on the right. HDBSCAN detects 42 clusters with an outliers class (-1) identified by colors.

The clusters obtained in 3 dimensions by HDBSCAN displayed in Figure 3.19a overlap to separate them, and the frequency and pulse width averages are computed from each cluster to obtain a new representation highlighted in Figure 3.19b. Each dot represents the clusters' frequency and pulse width averages. This new representation improves cluster separation and avoids overlapping cluster pulses.

At this step, each cluster is represented by a frequency and pulse width averages: $\overline{f_n}$ and $\overline{pw_n}$. This new representation applies a classical hierarchical agglomerative clustering using Euclidean distances to group clusters with the same characteristics from frequency. The aggregations are presented in Figure 3.20. The pulse width is set aside because measurement error can significantly impact this feature, directly influencing the hierarchical agglomerative clustering results. As previously, Algorithm 11 is applied, and fusions are stopped using the Kolmogorov-Smirnov test, checking that the frequency and pulse widths of the clusters follow the same distribution. Alternatively, the Student test [Stu08] can also be used.

Analogously to the previous section, Figure 3.21 illustrates the use of optimal transport distances as an indicator to quantify similarity and dissimilarity between clusters. The frequency distributions of Clusters 3, 4, and 28 are plotted. Clusters 3 and 4 correspond to the same emitter, producing low transport costs as they transmit on the same frequency. In contrast, Clusters 4 and 28

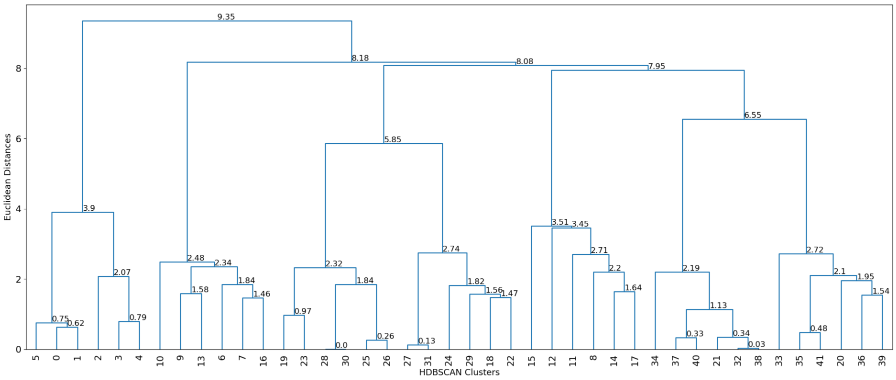


Figure 3.20: Dendrogram representing the aggregations at each iteration with the logarithm Euclidean distances values displayed.

are characterized by higher transport costs due to the different frequencies.

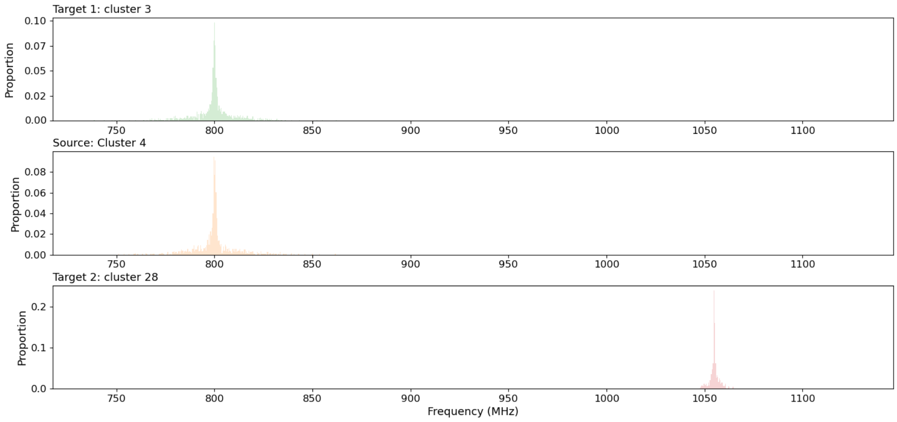
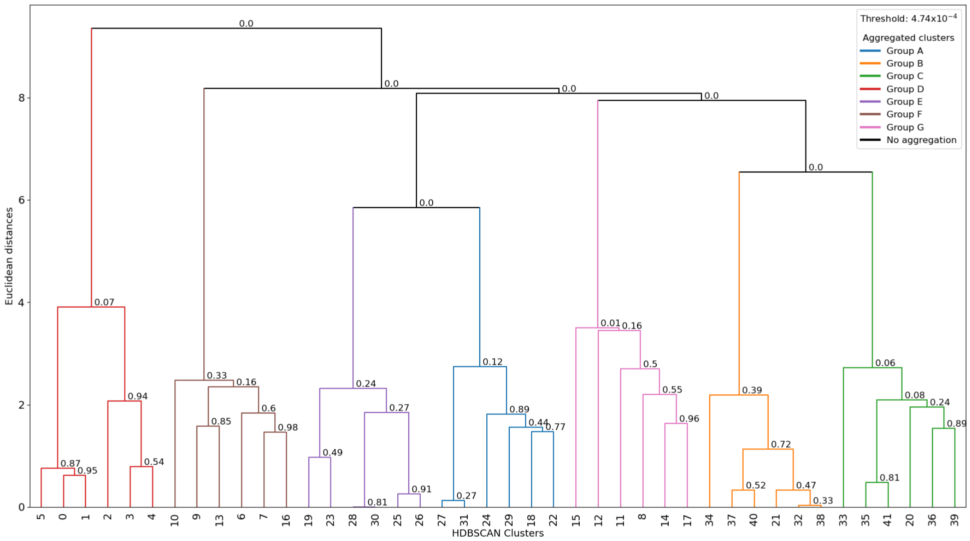
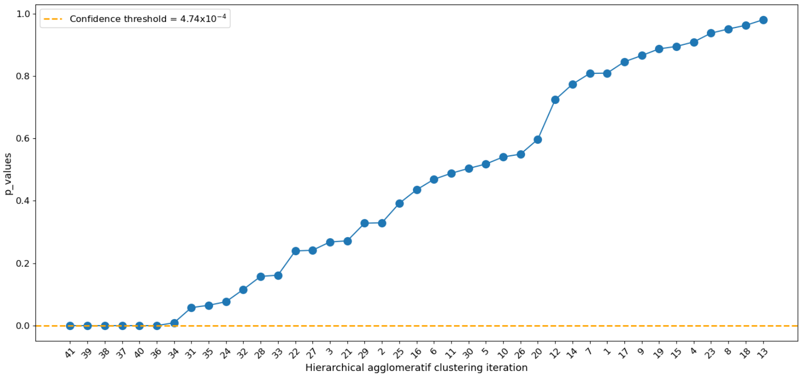


Figure 3.21: Frequencies distributions of clusters 3, 4 and 28.

The Figures 3.22b, 3.23b and 3.24b present the p-values obtained at each iteration of the HAC according to Kolmogorov-Smirnov, Epps-Singleton and the Student tests; the values are sorted to determine a break-point, highlighted by the orange line to set the confidence level. The idea is to identify the first non-zero p-values and the first break in the evolution of the p-values. Note that large breaks appear later in these figures. The values obtained are $\alpha_{ks} = 4.74 \times 10^{-4}$ for the Kolmogorov-Smirnov test, $\alpha_{epps} = 2.10^{-4}$ for the Epps-Singleton test, and $\alpha_{std} = 5.85 \times 10^{-4}$ for the Student test. Based on these results, the top plots show the dendrograms, with each color representing an aggregated cluster obtained following the fixed confidence level of each test. All the different tests led to the same grouping, and the decision models identified seven aggregated clusters.

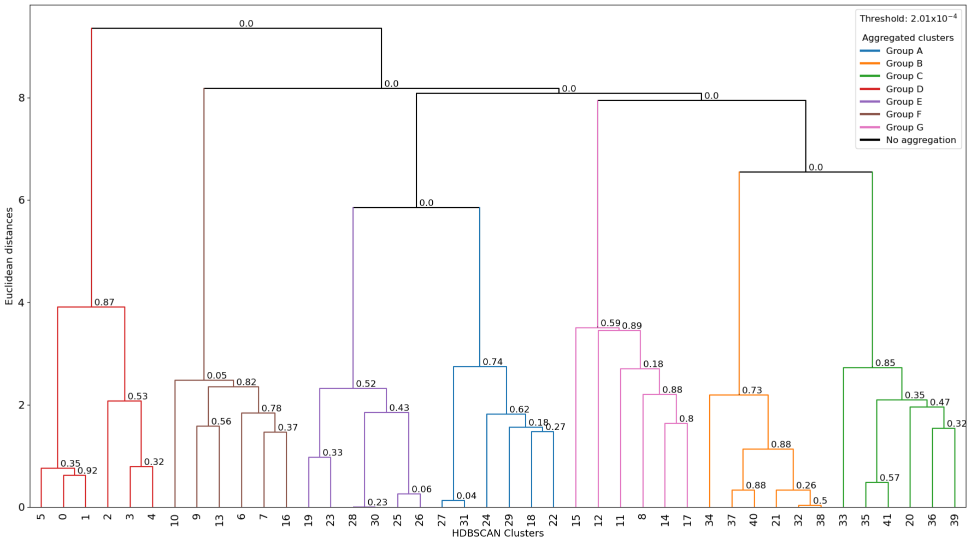


(a) Dendrogram.

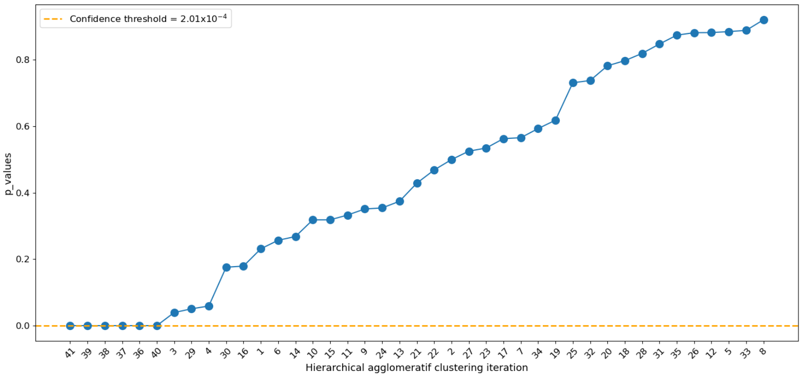


(b) Elbow tricks.

Figure 3.22: Hierarchical agglomerative clustering results according to the Kolmogorov-Smirnov test used with the orange line highlighting the estimated confidence level on the bottom plot.

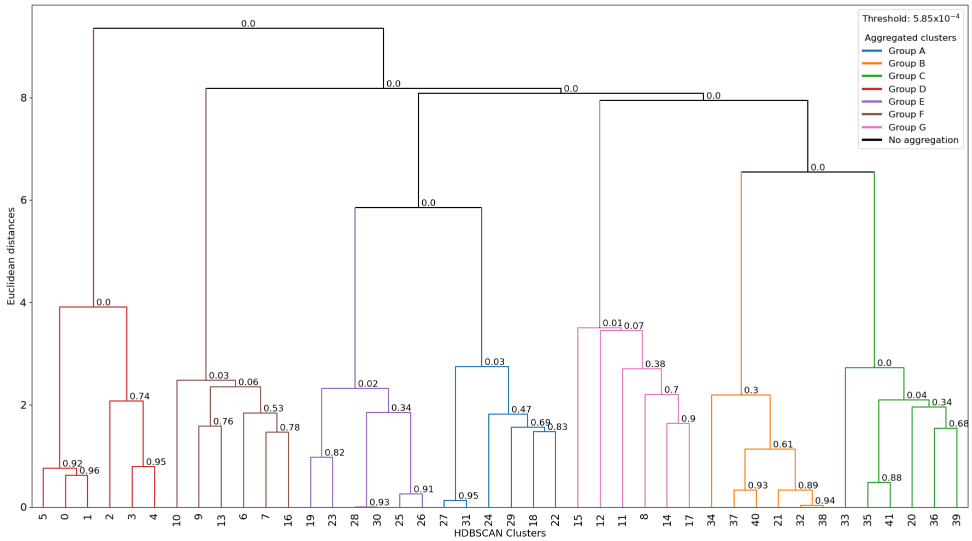


(a) Dendrogram.

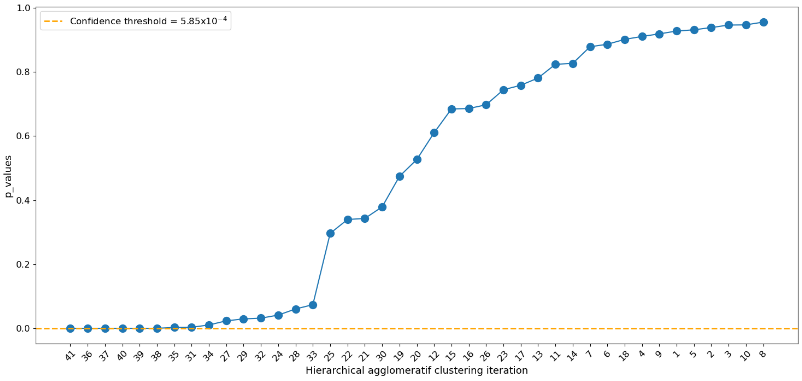


(b) Elbow tricks.

Figure 3.23: Hierarchical agglomerative clustering results according to the Epps-Singleton test used with the orange line highlighting the estimated confidence level on the bottom plot.



(a) Dendrogram.



(b) Elbow tricks.

Figure 3.24: Hierarchical agglomerative clustering results according to the Student test used with the orange line highlighting the estimated confidence level on the bottom plot.

From these results, the clusters were aggregated into 7 groups and are represented in Figure 3.25. Emitters having characteristics of agility, emitting on several frequencies, are always represented by several clusters. The Algorithm 8 is now applied to perform the last groupings from these clusters.

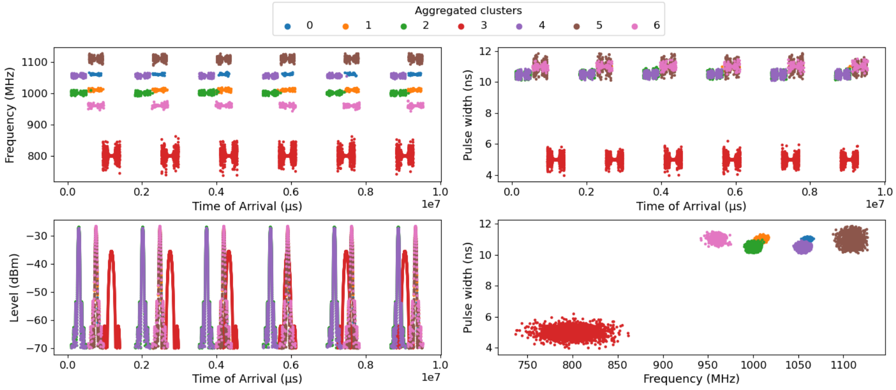


Figure 3.25: Aggregated clusters obtained following the hierarchical agglomerative clustering using Euclidean distances pruning with statistical test. Each color represents an aggregated cluster.

3.2.4 . Cluster fusion with hierarchical agglomerative clustering using optimal transport distances

This last step of the algorithm is similar to the last step of the previous method in Section 3.1, based on hierarchical clustering with optimal transport distances. Figure 3.26 presents the final three sets of pulses identifying and an outlier class (-1). The methodology could separate and regroup emitters' pulses with similar characteristics in the presence of noise; HDBSCAN excluded about 9% of pulses. Excluding these points isn't problematic since there are enough classified points to extract the characteristics of each cluster, resulting in excellent performance during the identification phase.

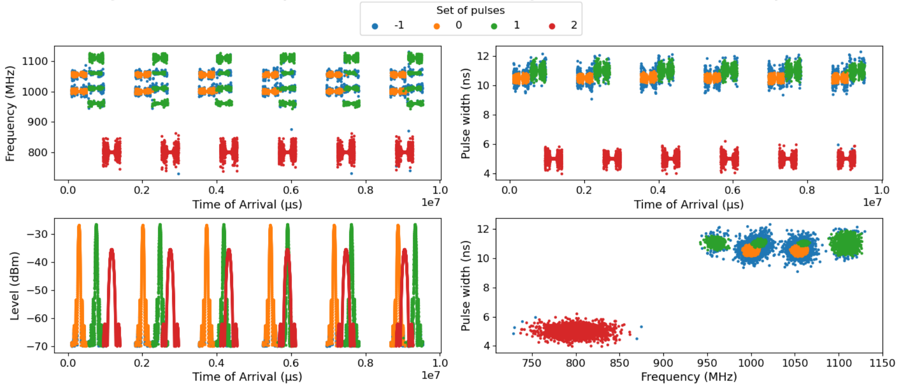


Figure 3.26: Final grouping after applied hierarchical agglomerative clustering combined with optimal transport distances. Each color identified a set of pulses with an outliers class (-1).

3.2.5 . Computational complexity

The complexity of this second method is the sum of the complexity of HDBSCAN, the pre-cluster aggregation, and the cluster aggregation. The complexity of HDBSCAN and the cluster aggregation is identical to the first method, with C now being the number of pre-aggregation clusters. The pre-cluster aggregation is a simple hierarchical clustering, with complexity $O(P^3)$, with P the number of clusters identified by HDBSCAN. P can be estimated by $P = CTS$, where S is the average number of sweep by second. Compared to the previous algorithm, the complexity is augmented by $O(C^3T^3S^3)$, or $O(\bar{C}^3K^3T^3S^3)$.

To conclude, this chapter presented two methods to deinterleave a RADAR signal. The first method, HACOT, in 2 steps, uses the HDBSCAN clustering algorithm to separate pulses based on frequency and pulse width. As several clusters can represent a RADAR, a hierarchical agglomerative clustering using optimal transport distances from the time of arrival and the level was developed to group these clusters. The second method, named IHACOT, includes an additional step. The pulses are first separated with HDBSCAN based on time of arrival, frequency, and pulse width. The pulses were first grouped using hierarchical agglomerative clustering based on frequency before applying the previously developed hierarchical agglomerative clustering using optimal transport distances from the time of arrival and the level. The results are represented hierarchically, and two decision models are used to determine where to stop the aggregations proposed. The first is based on unsupervised statistical metrics providing a unique pruning threshold, and the second is based on statistical tests allowing personalized pruning to be obtained. Several signals were used to illustrate the different methodologies.

4 - Identification of RADAR from a set of pulses

Upon completion of the deinterleaving phase, when the pulses have been correctly isolated into distinct sets of pulses, the second step in the RADAR recognition process can proceed with the emitters identification. This chapter presents a new supervised methodology to identify emitters in an intercepted signal with the optimal transport distances by modeling signal missing pulses. The method developed is presented to identify the emitters by comparing sets of pulse characteristics to those of a database, gathering the typical emitters' references using only three features: frequency, pulse width, and the emission pattern (i.e., the pulse repetition of intervals). When intercepting a signal, some pulses may be missed (non-interception due to the captor, poor deinterleaving, noise...), leading to poor reconstruction of the emission pattern from the signal. Therefore, it is essential to model these errors during the identification phase. Finally, an application to the RADAR emitters classes used to simulate the signals is presented; the optimal transport distances are used to provide a risk analysis between the classes of RADAR emitters composing our identification database to quantify their proximity and provide additional information regarding the certainty of the results of the identification algorithm to the ESM operators.

4.1 . Emitters database representation

4.1.1 . Emitters classes characteristics

The initial step of the identification consists of representing the emitters' classes. The emitters database used to make the classification includes references from a large number of different types of emitters. Several characteristics describe an emitter, as shown in Section 2.3. In RADAR signal processing, particularly for military applications, access to information about RADARs can be challenging or impossible to obtain. Creating a database of RADAR characteristics can be complex. The objective is to propose a method that requires few information on RADARs. In this work, only three features are considered:

- Frequency, f_n
- Pulse width, pw_n
- Pulse repetition period, pri_n

Emitters have varying complexity and sophistication; the simplest emitters transmit continuously on a single frequency. In this scenario, a single class describes the emitter within the database. Recent emitters present more complex and similar electromagnetic spectra, exemplified by frequency agility characteristics or multiple operation modes: several frequencies, pulse widths, or speeds characterize them. Multiple classes, therefore, describe them in the database. The methodology can identify the emitter and the sub-operation modes employed by the emitters.

Emitter	**Frequency (MHz)**	**Pulse Width (ns)**	**PRI (μs)**	**PRI pattern**
A	1025, 1050 1075, 1100 1125, 1150	2	76, 77, 78, 79 80, 81, 82, 83 84, 85, 86, 87 88, 89, 90, 91 92, 93, 94, 95 96, 97, 98, 99	1, 1, 1, 1 1, 1, 1, 1 1, 1, 1, 1 1, 1, 1, 1 1, 1, 1, 1 1, 1, 1, 1
B$_1$	972	1.2	100	6
B$_2$	972, 1072 1112	4	110	8
C	1000	3	82, 110	1, 3
D	825, 884	2.9	85, 86, 87, 88 89, 90, 91, 92 93, 94, 104, 106 108, 110	1, 1, 1, 1 1, 1, 1, 1 1, 1, 1, 1 1, 1
E	1125	5	65	4
F	800	3.7	80, 85, 90, 95 100, 105, 110 115, 120	2, 2, 2, 2 2, 2, 2 2, 2
G	1125	5	70	1
H	938	2.3	Min: 75, Max: 95	1
I	858	4.6	Min: 65, Max: 85	1

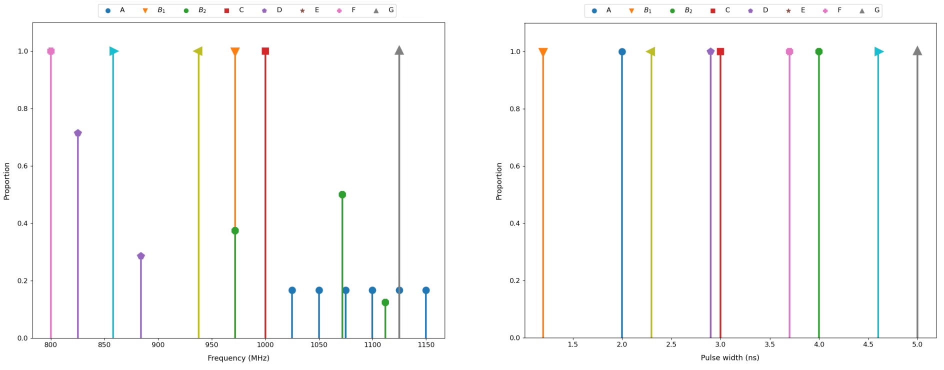
Table 4.1: Simulated emitter characteristics.

Table 4.1 groups the characteristics of the simulated emitters classes used to illustrate the methodology. Some emitters have simple characteristics, such as Emitter E, which transmits on a single frequency, pulse width, and PRI. In contrast, more complex emitters are characterized by several frequencies as Emitter A or by a random PRI as emitters H and I. Emitters B$_1$ and B$_2$ are the two operating sub-modes of Emitter B, which are characterized by two different pulse widths at different speeds. In the database, this emitter is represented by two different classes.

The database combines the characteristics of simple emitters such as Emitters I or H as shown in Figures 4.1a and 4.1b, which are single-frequency, in contrast to Emitter A, which has agility characteristics and transmits on six different frequencies. Frequency bands can also be shared, as shown by Emitters A and G. Emitters E and G are inseparable from this plane because they transmit on the same frequency and share one of the frequencies of Emitter A. Some emitters emit very different pulse widths, such as Emitters B_1 and G. They are easily distinguishable. In contrast, Emitters E and G share the same pulse widths and cannot be differentiated from this characteristic. Generally, emitters transmit on pulse width, making this feature insufficient to differentiate them.

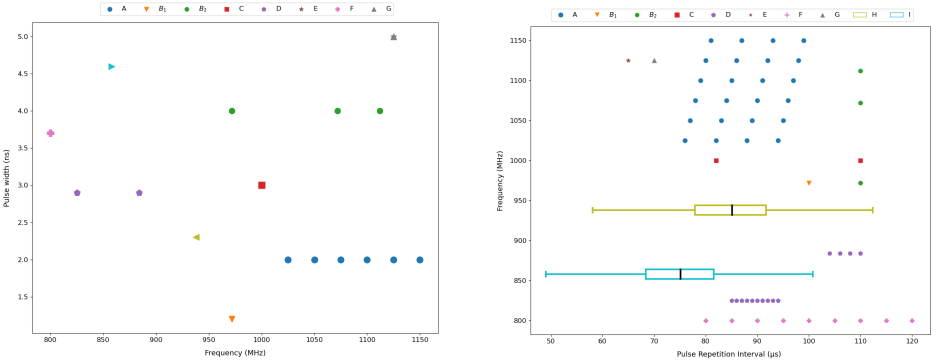
Using only one feature is insufficient to discriminate emitters; Combining several features leads to better discrimination, as highlighted in Figures 4.1c and 4.1d. Emitters A and G are now spaced out in the plane and easily identifiable but still insufficient for Emitters E and G. As previously explained in Section 2.3, the PRI is highly discriminating for emitters due to its frequent uniqueness. As previously explained in Section 2.3, there are two operating modes concerning the PRI of RADARs; one has the RADARs being characterized by a deterministic temporal emission pattern such as RADARs A or F, and those with a random temporal emission pattern such as RADAR H. In this case, PRI values are randomly generated from an interval.

Figure 4.1e illustrates the random PRI distribution of Emitter H represented by a Gaussian distribution. This representation simplifies this emitter type's data handling and integration into the identification algorithm. When a Gaussian represents random PRI emitters, their characteristics in 3 dimensions could be visualized in Figure 4.1f. Emitters E and G are separated in the plan but remain close due to their similar PRIs. Using PRI-based algorithms when the signal is poor quality or poorly estimated could lead to misidentification.



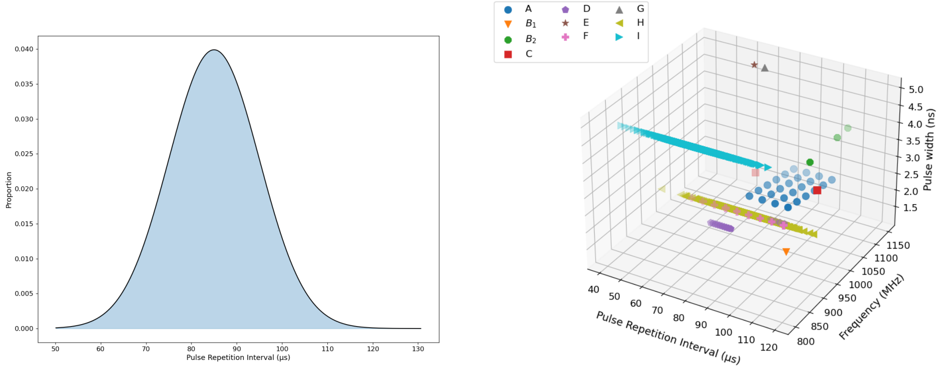
(a) Frequency plane.

(b) Pulse Width plane.



(c) Frequency x Pulse Width plane.

(d) Pulse Repetition of Interval x Frequency plan.



(e) Random PRI of Emitter H.

(f) PRI x Frequency x Pulse Width plan

Figure 4.1: Representation of emitters pulses parameters.

4.1.2 . Missing pulse rate modeling of emitters

As a reminder, the identification methodology is built from 3 features, including the pulse repetition of intervals. As explained in Section 2.4, when a signal groups the mixed pulses of different emitters, two successive pulses are not guaranteed to belong to the same RADAR (poor estimation of the signal, inadequately deinterleave pulses, or identified as outliers), making it impossible to obtain the correct pri distribution of the emitter from the signal. To make sense, the pri must be calculated from the pulses belonging to the same emitter. Therefore, a different distribution is obtained: the $dtoa$ (also named the estimated pri).

Therefore, the missing pulse rate must be considered during the identification phase when representing the pri distribution from the emitter database characteristics to make a coherent comparison with the $dtoa$ distribution resulting from the signal. To solve this problem, a new method based on the total probability formula is used, illustrated in Algorithm 14, to represent this phenomenon:

$$\mathbb{P}(dtoa = t) = \sum_{n \in \mathbf{R}_+} \mathbb{P}(dtoa = t | N = n)\mathbb{P}(N = n), \tag{4.1}$$

with N the discrete random variable representing the number of the missing pulses such that $\mathbb{P}(N = n) = p_n$ for $n \in \mathbf{N}_+$ represents the distribution of the missing pulses, and $\mathbb{P}(dtoa = t|N = n)$ the $dtoa$ distribution conditionally to the number of missing pulses. Let us now represent the distribution of the number of missing pulses.

Algorithm 14 Pulse Repetition of Interval modeling process according to the missing pulse rate - PriModLost

Data : PRI values from emitters database, T

Parameters : Missing pulse rate, α

Procedure :

1. Representation of the missing pulses number distribution according to α: $\mathbb{P}(N = n) = (1 - \alpha)\alpha^n$
2. Representation of the $dtoa$ distribution for a fixed number of missing pulses n according to the PRI process: $\mathbb{P}(dtoa = t|N = n)$:
   - **Discrete**:
     $$\mathbb{P}(dtoa = t|N = n) = \frac{1}{K}\sum_{k=0}^{K-1} \delta_{t_{nk}}$$
   - **Random**:
     $p_{nk}(t)$ the probability density of T_{nk}
3. Representation of the $dtoa$ distribution conditionally to the number of missing pulses: $\mathbb{P}(dtoa = t)$:
   - **Discrete**:
     $$\nu_\alpha = \frac{1}{K}\sum_{n=0}^{\infty}(1 - \alpha)\alpha^n \sum_{k=1}^{K} \delta_{t_{nk}}$$
   - **Random**:
     $$p_\alpha(t) = \frac{1}{K}\sum_{n=0}^{\infty}(1 - \alpha)\alpha^n \sum_{k=1}^{K} p_{nk}(t)$$

Result : Set of PRI distribution according to different missing pulse rates, $dtoa$

Distribution of the missing pulses number: $\mathbb{P}(N = n)$

As a reminder, N is the discrete random variable representing the number of missing pulses. Assuming that the losses are independent, a geometric distribution can represent them:

$$\mathbb{P}(N = n) = (1 - \alpha)\alpha^n, \tag{4.2}$$

with α, the missing pulse rate. Now, one have to continue to represent the pri distribution conditionally on the missing pulse number.

Distribution of $dtoa$ for a fixed number of missing pulses n: $\mathbb{P}(dtoa = t|N = n)$

As previously explained, emitters can have several operating modes for their pri. The distinction between these two operating modes represents the distribution of $dtoa$ when the missing pulse number n is fixed. Let K be the number of pri of the emitters, one have:

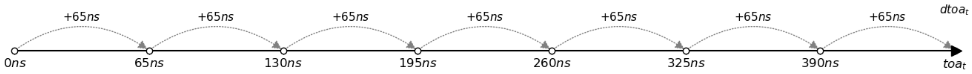
- **Discrete process:** the sum of successive pri starting with the k^{th} value of pri is given by the following expression:

  $$t_{nk} = \sum_{m=k}^{k+n} pri_{m,mod[K]}. \tag{4.3}$$

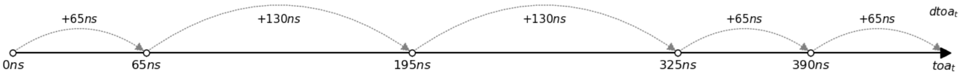
  Assuming that the values of pri have uniform probabilities of appearance, the conditional distribution of $dtoa$ for n fixed is given by

  $$\mathbb{P}(dtoa = t|N = n) = \frac{1}{K} \sum_{k=0}^{K-1} \delta_{t_{nk}}. \tag{4.4}$$

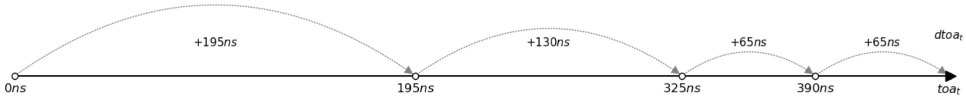
  For example, consider Emitter E from Table 4.1, characterized by one PRI (65ns). When the signal is well measured, and all the pulses are intercepted, ***e.g.***, $\alpha = 0\%$, the $dtoa$ distribution is composed of a unique value 65 ns with a probability equal to 1 as illustrated in Figure 4.2a. When 25% or 45% of the pulses are missing, other pri values appear and are obtained as the multiples of the initial pri value, and their probabilities are given by using Equation (4.8) and illustrating in Figures 4.2b and 4.2c.



(a) $\alpha = 0\%$.



(b) $\alpha = 25\%$.



(c) $\alpha = 45\%$.

Figure 4.2: Example of the $dtoa$ values according to different missing pulse rate of Emitter E.

When the emitter has a more complex transmission time pattern as Emitter C, which is characterized by the two PRI 82 and 110 ns with the following pattern $pri = [82, 110, 110, 110]$. The PRI values are obtained by analyzing the successive PRI values starting with each PRI value of the transmission time pattern as illustrated in Figure 4.3.

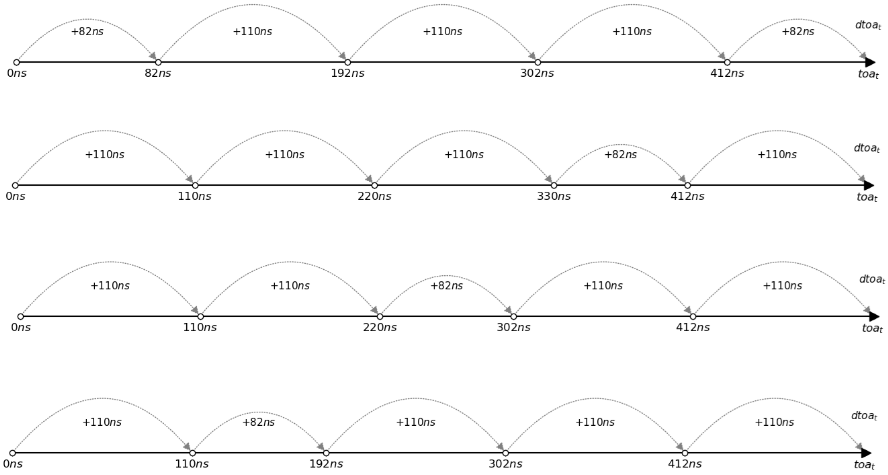
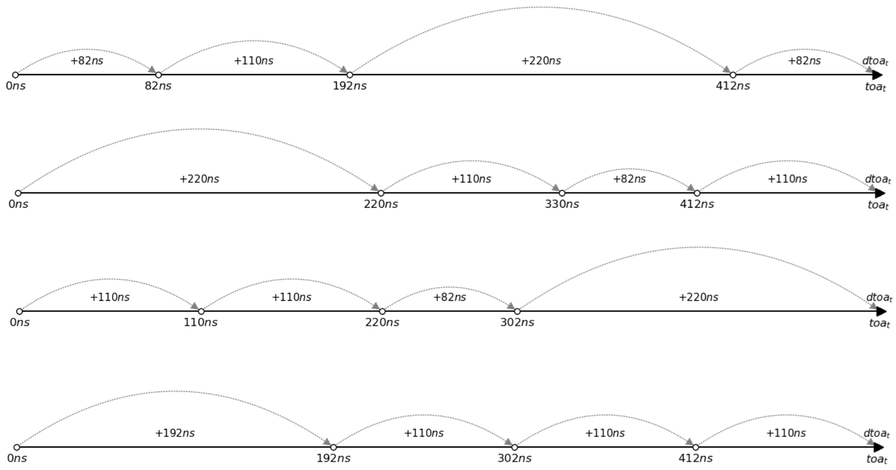
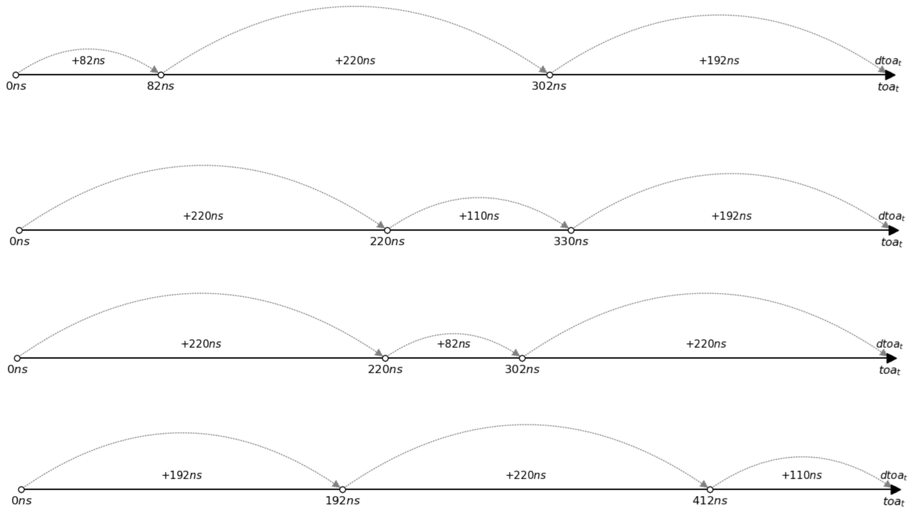


Figure 4.3: Emitter C distribution with $\alpha = 0\%$ according to each PRI values.

Then, considering a given missing pulse rate, other PRI values appear similarly to the Figure 4.2 in each pattern starting by the k^e PRI values of the emitters as highlighted in Figures 4.4.



(a) $\alpha = 25\%$.



(b) $\alpha = 45\%$.

Figure 4.4: Example of the distribution of the $dtoa$ according different missing pulse rate of Emitter C.

- **Random process:** the pri is represented by a Gaussian random variable J_m with mean pri_m and variance σ^2 such that:

$$J_m \sim \mathcal{N}(pri_m, \sigma^2). \tag{4.5}$$

  Analogously to Equation (4.3), one have:

$$T_{nk} = \sum_{m=k}^{k+n} J_m. \tag{4.6}$$

  Assuming that the J_m are independent, the distribution of the $dtoa$ is given by:

$$T_{nk} \sim \mathcal{N}(t_{nk}, (n+1)\sigma^2). \tag{4.7}$$

  Thereafter, $p_{nk}(t)$ refers to the probability density function of T_{nk}.

Representation of the $dtoa$ distribution conditionally to the number of missing pulses: $\mathbb{P}(dtoa = t)$

- **Discrete process:** by applying Equations (4.1), (4.2) and (4.4) one have:

$$\nu_\alpha = \frac{1}{K}\sum_{n=0}^{\infty}(1-\alpha)\alpha^n \sum_{k=1}^{K} \delta_{t_{nk}}. \tag{4.8}$$

- **Random process:** by applying Equations (4.1), (4.2) and (4.7) one have:

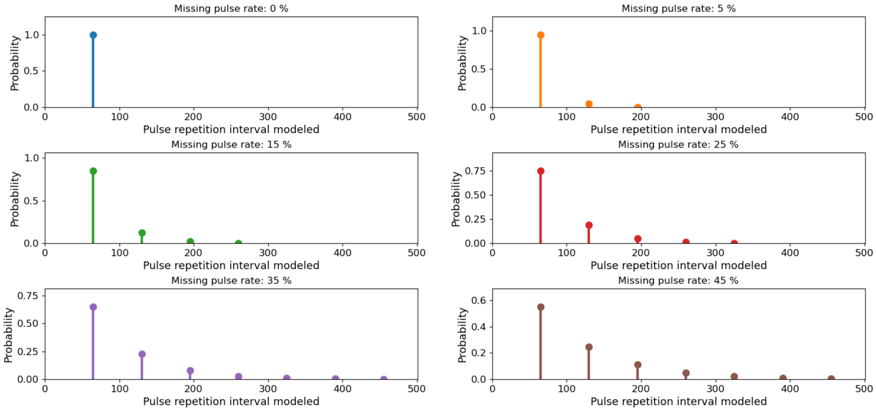
$$p_\alpha(t) = \frac{1}{K}\sum_{n=0}^{\infty}(1-\alpha)\alpha^n \sum_{k=1}^{K} p_{nk}(t), \tag{4.9}$$

  where K is the number of pri of the considering emitter, p_{nk} is the density of the Gaussian variable T_{nk} and T_{nk} the law of the $dtoa$.

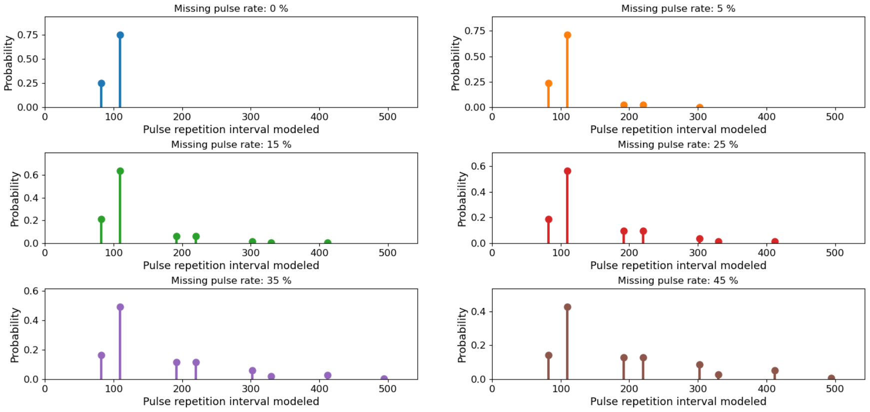
The discrete PRI process is a particular case of a random PRI process with a variance equal to 0.

Theoretically, α can be set between 0 and 1, but in practice, α is fixed between 0 and 50%. Multiple analyses have been conducted, and it has been observed that beyond the 50% of missing pulses, the intercepted signal lacks sufficient reliable information, is complex to analyze, and the modeled distributions tend to be similar. In the following, the modeled distributions have been shortened, and only some values of PRI have been selected to facilitate the interpretation.

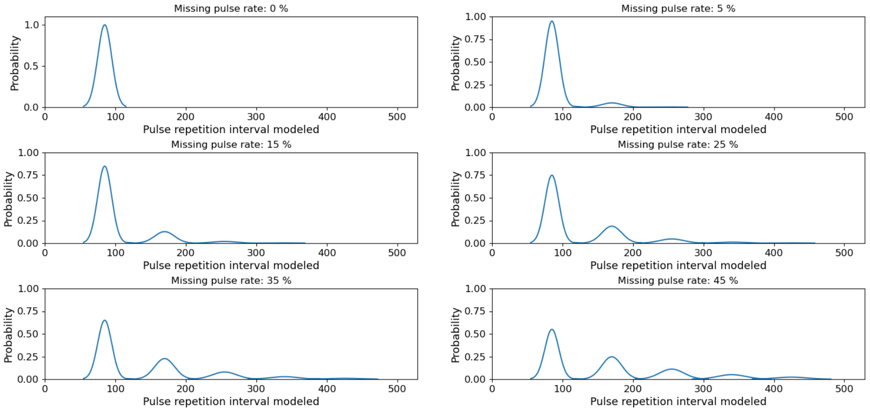
The missing pulse modeling method represents the PRI distributions of each emitter, as illustrated in Figures 4.5. Now, each emitter is represented by several distributions considering the phenomenon of missing pulses.



(a) Emitter E.



(b) Emitter C



(c) Emitter H.

Figure 4.5: Modeling of the $dtoa$ distribution of emitters according to different missing pulse rates.

4.2 . Decisional model based on optimal transport

The identification methodology, named IDOT, is outlined in Algorithm 15. The methodology is based on building a distance between a set of pulses and emitter classes through optimal transport to identify the emitters from frequency, pulse width, and temporal emission pattern.

Algorithm 15 Identification of Emitters using Optimal Transport Distances - IDOT

Data :

- Database, T grouping characteristics of emitters, described by:
    - Pulse repetition interval, pri_n
    - Frequency, f_n
    - Pulse width, pw_n

  With n, the number of transmissions of the characteristic of the emitter.
- Set of pulses X, described by:
    - Frequency, f_m
    - Pulse width, pw_m
    - Time of arrival, toa_m

Parameters :

- Missing pulse rate, α

Procedure :

1. Construction of a measure describing each emitter class belonging to T based on pri_n, f_n, and pw_n according to α:
    (a) Modeling the emitter classes pri according to α with the modeling approach presented in Algorithm 14: $p_\alpha(t)$
    (b) Construction of a measure based on $p_\alpha(t)$, f_n, and pw_n: μ
2. Construction of a probability distribution from the deinterleaving set of pulses based on $dtoa_m, f_m, pw_m$: ν
3. Identification of the emitter class by assigning the closest emitter class in the optimal transport distance sense to the set of pulses: $j^\star = \operatorname{argmin}_{\mathbf{j} \in (1,\ldots,J)} d\left(\mu_j, \nu\right)$

Output : $j^\star$, identified emitter class.

After representing the frequencies, pulse widths, and the modeled pulse repetition of intervals of the emitters classes as a probability distribution, the set of pulses is also modeled from frequency, pulse width, and the differentiated time of arrival, as follows:

$$\nu = \frac{1}{M} \sum_{m=1}^{M} \delta_{dtoa_m, f_m, pw_m}, \tag{4.10}$$

with M, the number of pulses in the set. A set of pulses class is then assigned by identifying the closest emitter class in the optimal transport distance sense:

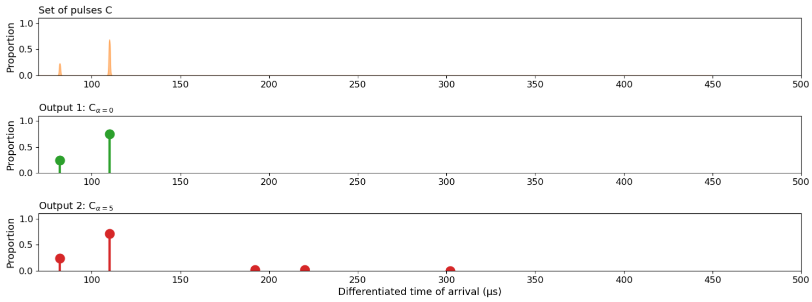
$$j^\star = \operatorname*{argmin}_{\mathbf{j} \in (1,\ldots,J)} d\left(\nu, \tau_j\right), \tag{4.11}$$

with J, the number of emitters classes. The algorithm identifies the emitter $j^\star$ as the true class. The received pulses are binned in frequency, pulse width, and pulse repetition intervals modelized to reduce the computation cost of computing the distances. With sufficiently small bin sizes, the perturbation of the distance is small enough so that the order of the distances between the data and the classes is conserved.

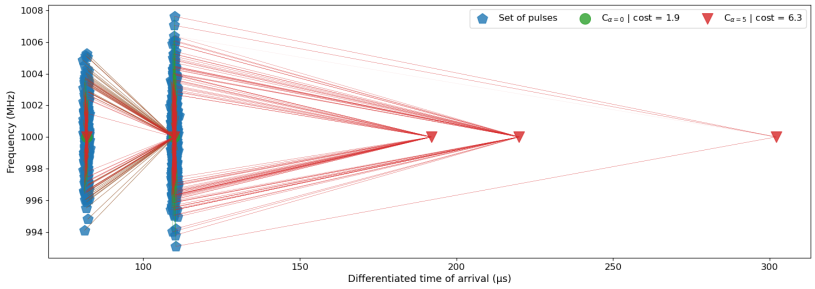
From the classes of emitters described in Table 4.1, a signal grouping the pulses of Emitter C was simulated by adding noise to illustrate the identification methodology. The classification database has been formatted to consider a missing pulse rate ranging from 0 to 45% and is therefore characterized by 60 classes. As a reminder, Emitter C is characterized by 2 PRI values: 82 and 110 ns, and the $dtoa$ is calculated as the time difference of successive pulses from this signal. Let us analyze the first scenario where all pulses have been intercepted; the distribution of the $dtoa$ is shown in the top plot in Figure 4.6, followed by the two first classes identified by the algorithm. The algorithm calculates the transport cost between the data distribution and the modeled classes and assigns the closest class in the optimal transport distance sense to the set of pulses. The data distribution and the first output distribution are similar; Both transmit at the same speeds, whereas several PRIs characterize the distribution of the second output, implying multiple displacements between points, increasing the transport cost.

The transport plans are displayed in two dimensions to simplify the interpretation of the results. The bottom plot in Figure 4.6 overlays the pulses and the two closest classes. It displays the transport plane used by the data distribution points, represented by the blue dots, toward the two first outputs, respectively, represented by the green and red dots. The characteristic points of the closest class correspond well to the data compared to the other class. The classifier correctly identifies the emitter class distribution according to the missing pulse rate in the data. The points displacements highlighted in red are sent in different directions across 250ns, increasing the optimal transport cost, whereas few displacements characterize the green points.

Similarly, a signal gathering the pulses of the Emitter C was simulated by losing 35% of the pulses and is presented on the top graph in Figure 4.7 and the two graphs below represent the distribution of the $dtoa$ of the two closest classes returned by the algorithm. The data distribution is similar to the distribution of the modeled PRI with a 35% loss below. The transport cost between these two distributions will be minimal, and the algorithm will assign this class to the data. This result is confirmed when analyzing the transport plans in the bottom plot in Figure 4.7; the green dots representing the emitter class distribution with 35% losses are spread along the graph and overlap the data dots; the points have few displacements to coincide. The algorithm identifies the emitter class in the data but also can estimate the missing pulses rate, indicating the reliability of the identification. To conclude, the proposed identification method is conclusive and can consider the underlying phenomenon of missing pulses when modeling the PRI.

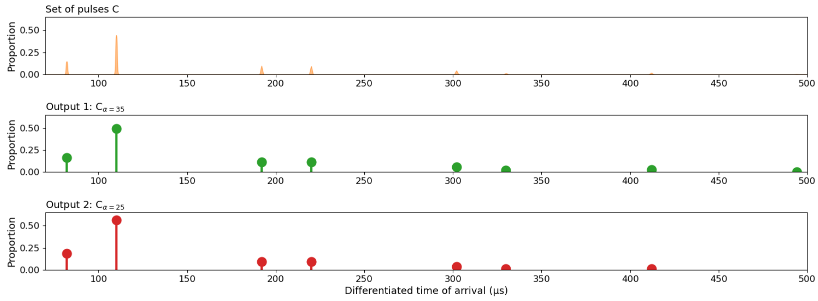


(a) Set of pulses and the two closest classes identified by the algorithm.

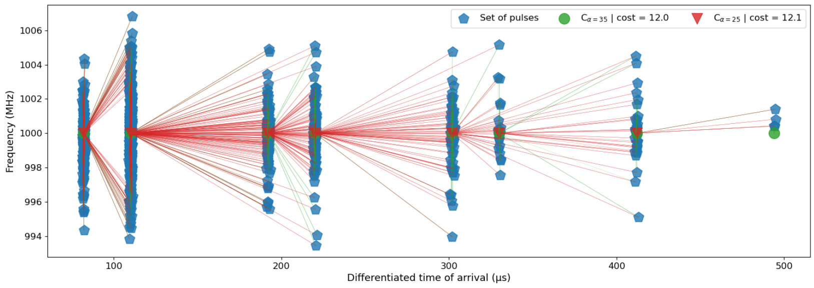


(b) Optimal transport plan between.

Figure 4.6: Identification outputs for Emitter C when all pulses are present.



(a) Set of pulses and the two first classes identified by the algorithm.



(b) Optimal transport plan.

Figure 4.7: Identification outputs for Emitter C when 35% of the pulses are missing.

4.3 . Computational complexity

The computational complexity of the identification algorithm can be estimated firstly in function of the number N of pulses in the set of pulses and the number of bins B.

Then, by estimating the transport cost using the Wasserstein distance [DGK18] between the set of pulses and the classes from the database grouping the emitters characteristics of J of classes.

According to [DGK18], the computational complexity between two datasets of size N can be estimated by $O(N^2)$. It can be reasonably assumed that the size of the pulse set is greater than any RADAR class in the database.

The transport cost complexity between the set of pulses and the J classes can be maximally approximated by $O(B^2 J)$.

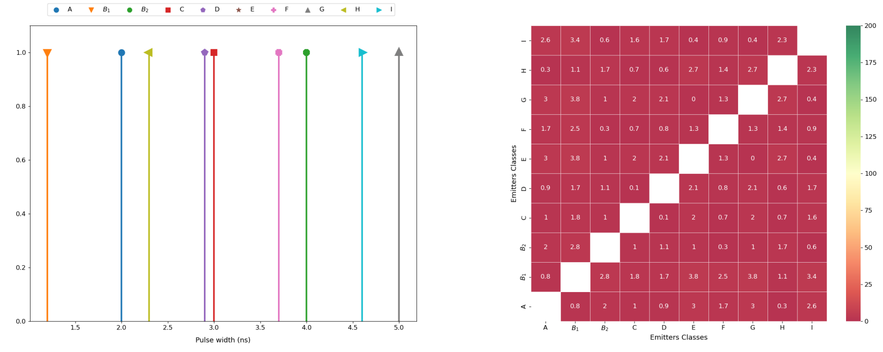
4.4 . Proximity Analysis between emitters

To complete the outcomes of the deinterleaving and identification analyses, an examination was carried out between the emitters with the optimal transport distances. Similar to Chapters 3 and 4, the optimal transport distances were used to measure the similarity between the classes of emitters and provide additional information to the ESM operators regarding the integrity of the results. The emitters presented in Chapter 4 are used to illustrate the methodology

The emitter's characteristics are represented in the left plots in Figures 4.8 through several planes. A distribution from the considered characteristics represents each class, and then the distance between them is calculated using the optimal transport; results are given in the right plots. Emitters H and I emitting through random PRI have been modelized with Gaussian distributions, and some of their pulses are displayed to facilitate visualization. The colorimetry of the cost matrices reflects the level of similarity between the emitters classes; red indicates a low transport cost between the emitters, meaning that the two classes share close characteristics and can be easily confused, while green indicates that the two classes are sufficiently different from being identifying minimizing the risk.

The pulse width cost matrix indicates that the emitters have very similar characteristics; the cost ranges between 0 to 3.4, specifically for Emitters G and E or even B_2 and G. These results are unsurprising because the emitters usually transmit on a single pulse width. The results do not show improvement with the PRI cost matrix since emitters share common transmission patterns. Usually, the PRI is distinct for each emitter, enabling their differentiation. In this example, the results are counter-intuitive because emitters with close PRI values were deliberately simulated to demonstrate the effectiveness of the identification method developed in Chapter 4.

The frequency is more discriminating, especially for emitters with several frequencies, and because emitters transmit through wide frequency bands. Combining several dimensions distinguishes better emitters, especially when using PRI, frequency, and pulse width. The pulse width provides no additional relevant information and could be neglected in the analysis.



Emitters Classes	A	B_1	B_2	C	D	E	F	G	H	I
I	2.6	3.4	0.6	1.6	1.7	0.4	0.9	0.4	2.3	
H	0.3	1.1	1.7	0.7	0.6	2.7	1.4	2.7		2.3
G	3	3.8	1	2	2.1	0	1.3		2.7	0.4
F	1.7	2.5	0.3	0.7	0.8	1.3		1.3	1.4	0.9
E	3	3.8	1	2	2.1		1.3	0	2.7	0.4
D	0.9	1.7	1.1	0.1		2.1	0.8	2.1	0.6	1.7
C	1	1.8	1		0.1	2	0.7	2	0.7	1.6
B_2	2	2.8		1	1.1	1	0.3	1	1.7	0.6
B_1	0.8		2.8	1.8	1.7	3.8	2.5	3.8	1.1	3.4
A		0.8	2	1	0.9	3	1.7	3	0.3	2.6

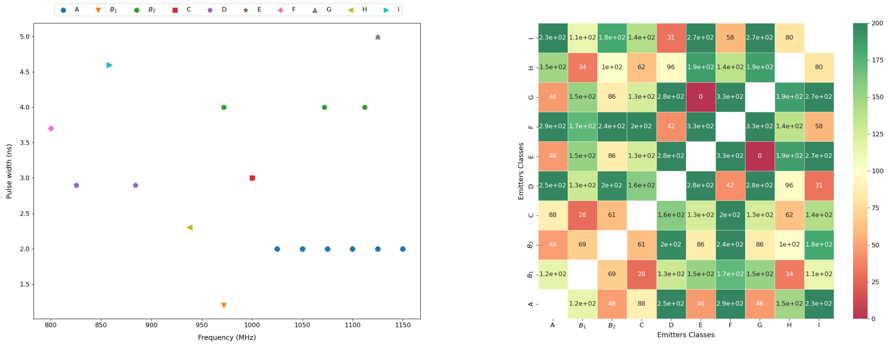
Characteristics. Cost matrix.

(a) Pulse width.

Emitters Classes	A	B_1	B_2	C	D	E	F	G	H	I
I	2.3e+02	1.1e+02	1.8e+02	1.4e+02	31	2.7e+02	58	2.7e+02	80	
H	1.5e+02	34	1e+02	62	96	1.9e+02	1.4e+02	1.9e+02		80
G	46	1.5e+02	86	1.2e+02	2.8e+02	0	3.2e+02		1.9e+02	2.7e+02
F	2.9e+02	1.7e+02	2.4e+02	2e+02	42	3.2e+02		3.2e+02	1.4e+02	58
E	46	1.5e+02	86	1.2e+02	2.8e+02		3.2e+02	0	1.9e+02	2.7e+02
D	2.5e+02	1.3e+02	2e+02	1.6e+02		2.8e+02	42	2.8e+02	96	31
C	88	28	60		1.6e+02	1.2e+02	2e+02	1.2e+02	62	1.4e+02
B_2	48	68		60	2e+02	86	2.4e+02	86	1e+02	1.8e+02
B_1	1.2e+02		68	28	1.3e+02	1.5e+02	1.7e+02	1.5e+02	34	1.1e+02
A		1.2e+02	48	88	2.5e+02	46	2.9e+02	46	1.5e+02	2.3e+02

Characteristics. Cost matrix.

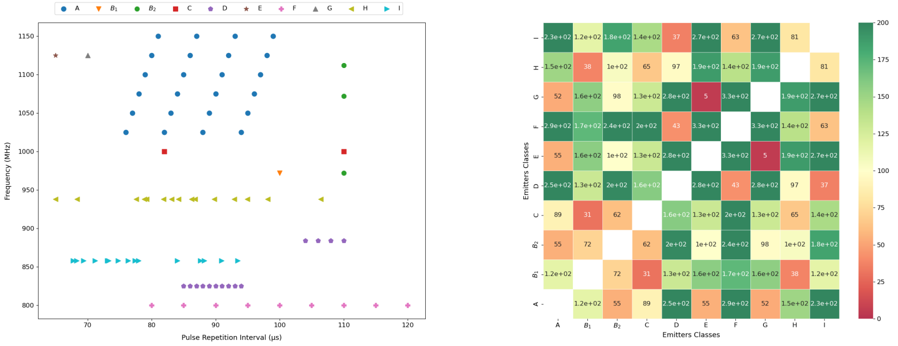
(b) Frequency.



Emitters Classes	A	B_1	B_2	C	D	E	F	G	H	I
I	2.3e+02	1.1e+02	1.8e+02	1.4e+02	31	2.7e+02	58	2.7e+02	80	
H	1.5e+02	34	1e+02	62	96	1.9e+02	1.4e+02	1.9e+02		80
G	46	1.5e+02	86	1.3e+02	2.8e+02	0	3.3e+02		1.9e+02	2.7e+02
F	2.9e+02	1.7e+02	2.4e+02	2e+02	42	3.3e+02		3.3e+02	1.4e+02	58
E	46	1.5e+02	86	1.3e+02	2.8e+02		3.3e+02	0	1.9e+02	2.7e+02
D	2.5e+02	1.3e+02	2e+02	1.6e+02		2.8e+02	42	2.8e+02	96	31
C	88	28	61		1.6e+02	1.3e+02	2e+02	1.3e+02	62	1.4e+02
B_2	48	69		61	2e+02	86	2.4e+02	86	1e+02	1.8e+02
B_1	1.2e+02		69	28	1.3e+02	1.5e+02	1.7e+02	1.5e+02	34	1.1e+02
A		1.2e+02	48	88	2.5e+02	46	2.9e+02	46	1.5e+02	2.3e+02

Characteristics. Cost matrix.

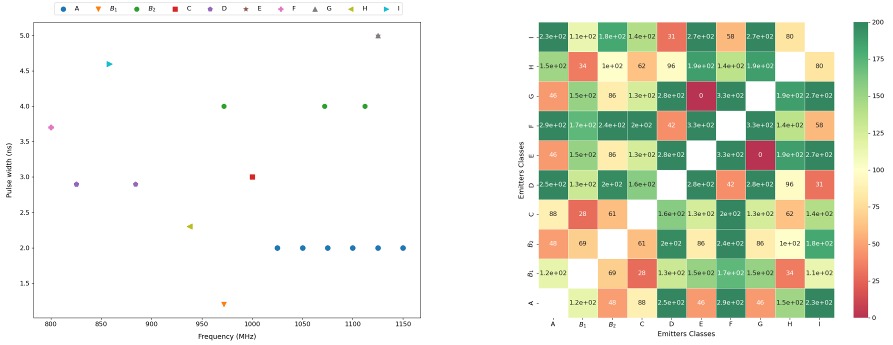
(c) Frequency x Pulse width.



Characteristics.

Cost matrix.

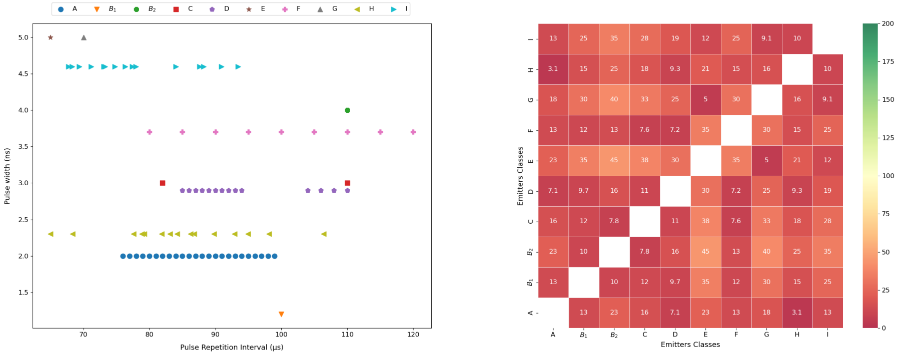
(d) Pulse Repetition Interval x Frequency.



Characteristics.

Cost matrix.

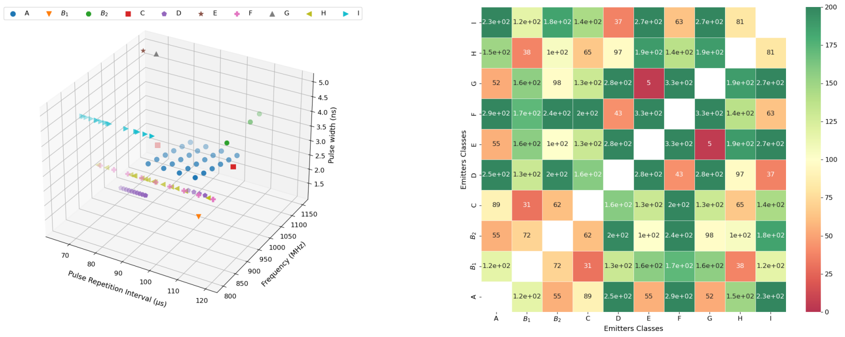
(e) Frequency x Pulse Width.



Characteristics.

Cost matrix.

(f) Pulse Repetition Interval x Pulse width.



	A	B_1	B_2	C	D	E	F	G	H	I
I	2.3e+02	1.2e+02	1.8e+02	1.4e+02	37	2.7e+02	63	2.7e+02	81	
H	1.5e+02	38	1e+02	65	97	1.9e+02	1.4e+02	1.9e+02		81
G	52	1.6e+02	98	1.3e+02	2.8e+02	5	3.3e+02		1.9e+02	2.7e+02
F	2.9e+02	1.7e+02	2.4e+02	2e+02	43	3.3e+02		3.3e+02	1.4e+02	63
E	55	1.6e+02	1e+02	1.3e+02	2.8e+02		3.3e+02	5	1.9e+02	2.7e+02
D	2.5e+02	1.3e+02	2e+02	1.6e+02		2.8e+02	43	2.8e+02	97	37
C	89	31	62		1.6e+02	1.3e+02	2e+02	1.3e+02	65	1.4e+02
B_2	55	72		62	2e+02	1e+02	2.4e+02	98	1e+02	1.8e+02
B_1	1.2e+02		72	31	1.3e+02	1.6e+02	1.7e+02	1.6e+02	38	1.2e+02
A		1.2e+02	55	89	2.5e+02	55	2.9e+02	52	1.5e+02	2.3e+02

Characteristics.

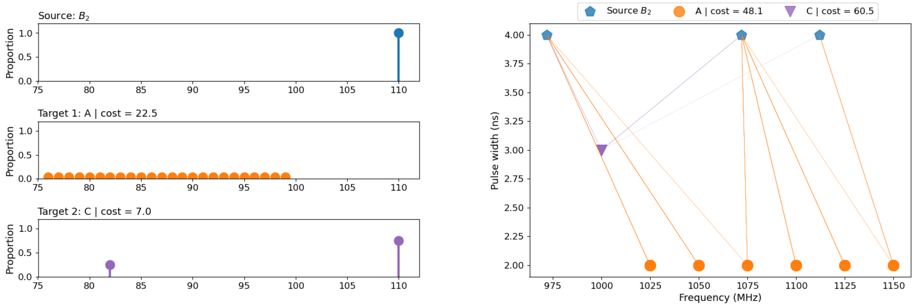
Cost matrix.

(g) Pulse Repetition Interval x Frequency x Pulse width..

Figure 4.8: Emitters characteristics on the left and the associated cost matrix computed on the right.

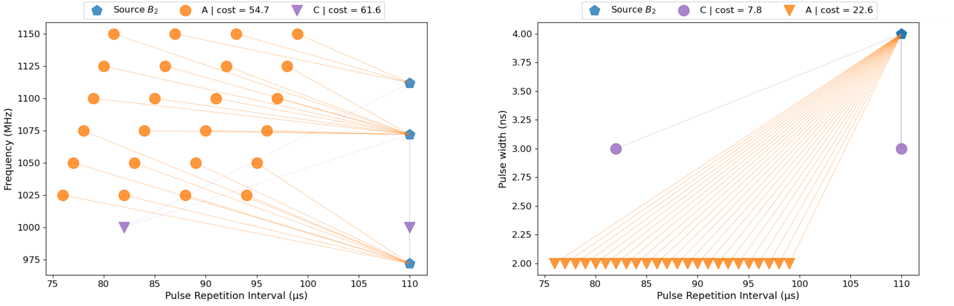
A focus on analyzing the proximity between Emitter B_2 toward Emitters A and C is provided in Figures 4.9. The PRI distributions of the three emitters are plotted in Figures 4.9a; each stem represents a PRI used by the emitter, and the height is the proportion of appearance of this PRI. The cost of moving B_2 PRI distribution is low due to a common PRI between the emitters; all points moved to a very close location, while the displacement cost between B_2 and A is three times higher as the points must make more significant displacements. The optimal transport plane from (pri_n, pw_n), (pri_n, pw_n), and (f_n, pw_n) are illustrated in Figures 4.9b, 4.9c, and 4.9d; the lines illustrate the movements of the B_2-characteristics points made, and the associated transport cost is displayed. The cost of transporting the characteristic points of Emitter B_2 toward C from the (pri_n, pw_n) plane is still low because the two emitters transmit a close pulse width (4ns for emitter B_2 against 5ns for Emitter C); all points being displaced to a nearby location. On the other hand, adding the frequency in the analysis separates these two emitters in the (pri_n, f_n) and (f_n, pw_n) planes. These analyses between the emitter classes provide the ESM operators with an additional indicator of the veracity of the results returned by the algorithms, leading to the identification of emitters that may be confused according to the features used.

The complete optimal transport planes between Emitters B_2 toward emitters classes are given in Figures 4.10.



(a) Pulse repetition interval distribution of emitters.

(b) Optimal transport plan from frequency and pulse width.

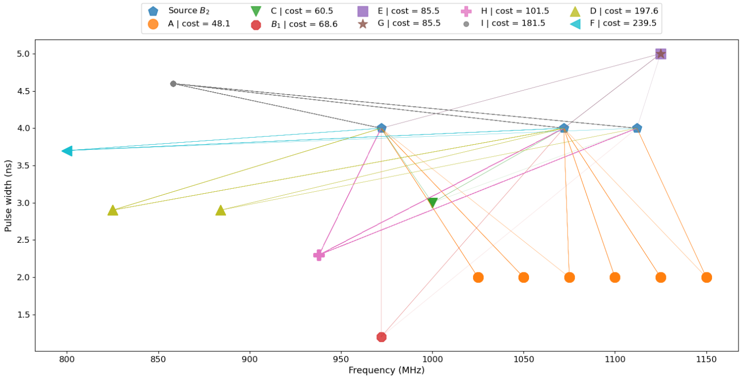


(c) Optimal transport plan from Pulse repetition interval and frequency.

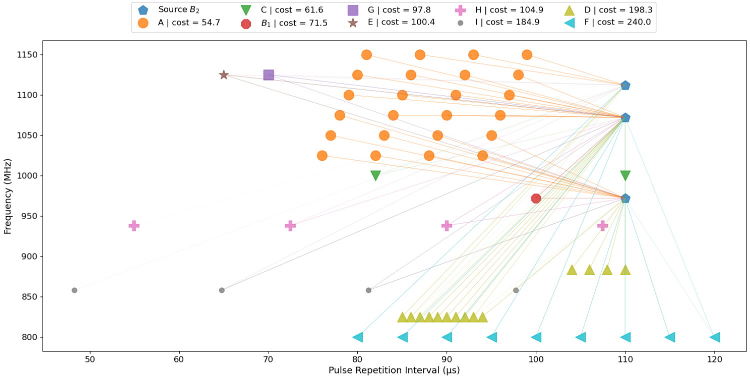
(d) Optimal transport plan from Pulse repetition interval and pulse width.

Figure 4.9: Focus on using optimal transport distances between Emitter B_2 toward Emitters A and C from pulse repetition interval, frequency, and pulse width.

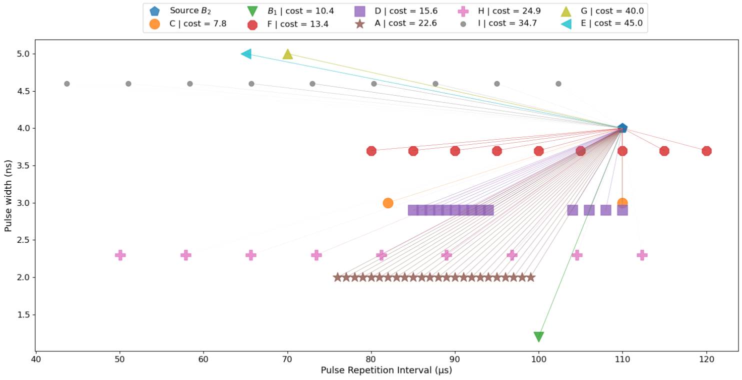
To conclude, this chapter presented a method for identifying a RADAR from a set of pulses based on three characteristics: frequency, pulse width, and temporal emission pattern. Before applying the identification algorithm on a set of pulses, a new method was proposed to take into account the missing pulses of the PRI. Indeed, to obtain the emission pattern of a RADAR, its pulses are differentiated in the time plane, giving the DTOA. However, this distribution may contain only some of the RADAR pulses, thus not allowing a reasonable reconstruction of the DTOA. The PRI was, therefore, modeled to take this phenomenon into account and make the identification method robust to missing pulses. Finally, optimal transport distances were used to analyze the proximity between emitters classes. The analysis of this proximity provides the ESM operators with a solid complement to the results' quality.



(a) Frequency and Pulse Width plane.



(b) Pulse Repetition Interval and Frequency plane.



(c) Pulse Repetition Interval and Pulse Width plane.

Figure 4.10: Optimal transport planes between Emitter B_2 toward other emitters from several planes.

5 - Performance evaluation

After presenting a complete RADAR Reconnaissance Process, composed of deinterleaving and identification phases, this chapter presents different methods to evaluate the performance of the proposed approach. First of all, the deinterleaving and identification algorithms are evaluated as two independent processes. Two parts are devoted to challenging the methods and highlighting their robustness. From a simulated signal, several parameters are manipulated to challenge the robustness of the proposed approaches. Then, several metrics are applied to evaluate and measure the performance of the algorithms. Generally, the identification is performed simultaneously with the deinterleaving step, but in this research, the deinterleaving and identification phases are presented as independent processes.

5.1 . Applying deinterleaving to realistic signals

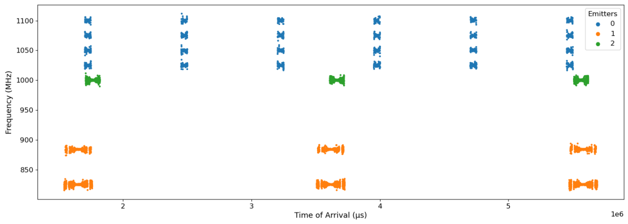
This section presents results to assess the quality of the proposed deinterleaving methods. The methods are tested on simulated signals grouping pulses of three emitters, with characteristics given in Table 5.1; The algorithms are compared to other deinterleaving methods before challenging them by playing on three parameters (outliers, noise, and missing pulses).

The signal is simulated with the developed simulator, and the receiver intercepted it with a detection threshold of -8 dBm by adding noise to make it realistic. The methods' performances are assessed using the Adjusted Rand Index (ARI) presented in Section 1.5.1 and the outliers rate to measure the part of pulses excluded from the process.

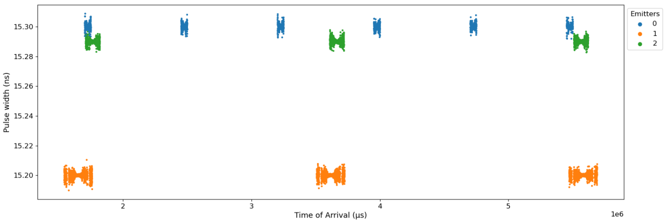
Name	**Frequency (MHz)**	**Pulse width (ns)**	**PRI (μs)**
0	1025, 1050, 1075, 1100	15.3	70, 110
1	825, 884	15.2	85, 86, 87, 88, 89 90, 91, 92, 93, 94, 104, 106, 108, 110.
2	860	15.3	95, 96, 97, 98, 99, 100.
Std	2.14	1e-3	2.55

Table 5.1: Simulated emitters characteristics.

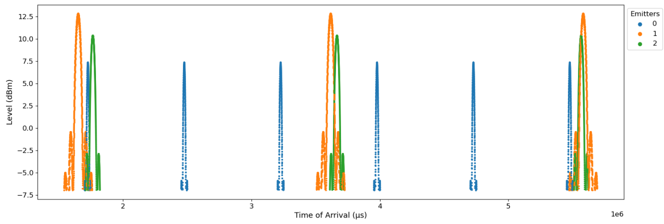
An example with 10044 pulses is displayed in Figure 5.1. The characteristics of the emitters, with multiple frequencies, similar pulse widths, and PRIs, are chosen to highlight the robustness of the methods concerning adverse cases, specifically the PRI-based methods. As the Emitters can emit on several frequencies with various pulse widths, methods only based on clustering fail to group all pulses of a given emitter as for emitters 0 and 1, who have their pulses spread out in several groups in the (f_n, pw_n) plane in Figure 5.1d.



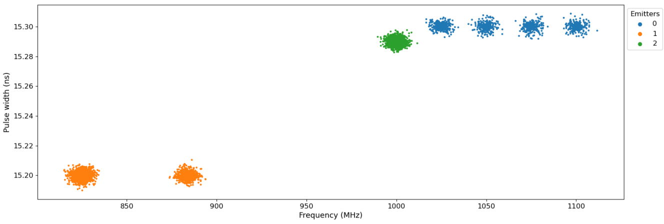
(a) Frequency x time of arrival plane.



(b) Pulse width x time of arrival plane.



(c) Level x time of arrival plane.

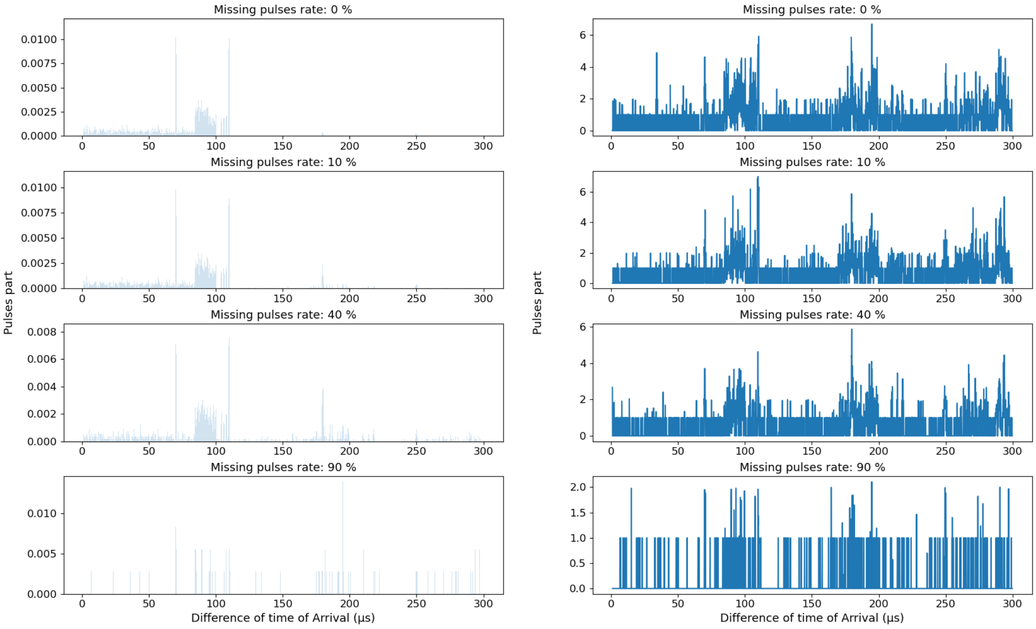


(d) Frequency x Pulse width plane.

Figure 5.1: Simulated signal gathering 10044 pulses from 3 emitters. Each color identified an emitter.

5.1.1 . Comparison with PRI-based methods

Two approaches were selected to be compared to the two deinterleaving methods developed in this work. One classic method, based on the analysis of PRI histograms [Mar89] and another, more recent, based on the PRI transform [Nis83], have been selected. Methods based on Deep Learning models have been deliberately set aside as they require a lot of data for the training step and perform inadequately with few data, while the proposed method is built in an unsupervised framework.



(a) Mardia output.

(b) PRI Transform output.

Figure 5.2: Outputs of the PRI-based methods to deinterleave the simulated signal according to different missing pulse rates.

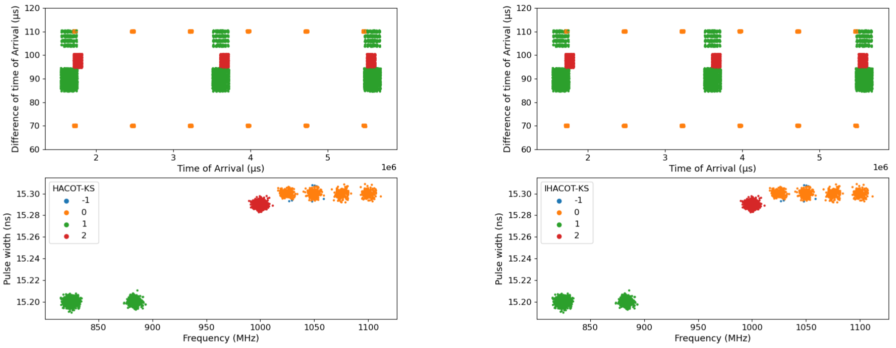
A longer signal was initially simulated, but only the 10044 successive pulses have been selected to facilitate alternative approach comparisons. Mostly, PRI methods are based on analyzing the linear combinations of possible PRI values, considerably increasing the computation time; these methods are more adapted to signals with few pulses. This is even more evident for PRI Transform's algorithm due to searching PRI multiples to analyze the combinations. Several ways have been considered to reduce the signal; randomly selecting values in the signal to reduce its size significantly affects the results of PRI-based methods. Since the proposed methods are not PRI-based, they are insensitive to random data deletion. The developed methodologies are much less sensitive to the signal size and can handle longer signals than PRI pattern-based methods.

Figures 5.2 represents the result of Mardia's and PRI Transform method applied to our previous signal according to different missing pulse rates. The first two plots present the results when all the pulses are present; both methods highlight a block of PRI oscillating around 70μs. A block of PRI between 80 and 110 μs is observed in Mardia outputs. The same results are observable in the PRI Transform method but are less pronounced. The application of the PRI Transform method is significantly time-consuming due to the length of the signal; the search for PRI values has been restricted to a predefined range instead of considering all time-of-arrival values. Even knowing the PRI values, the results are inconclusive in extracting the emitter's emission patterns. Even more, as the missing pulse rate increases, the readability and the extraction of these algorithms' results are distorted.

To facilitate understanding, HACOT and IHACOT identify the deinterleaving methods based on the 2D and 3D clustering developed in Section 3.1 and Section 3.2. The methods are performed based on applying the Kolmogorov-Smirnov test on all grouping phases to prune the dendrogram, and results are displayed in Figure 5.3. The top graphs plot the pulse repetition interval reconstructed from algorithm outputs, and the bottom graphs the (f_n, pw_n) planes. The emitters have different characteristics, and the pulses are sufficiently spread in the plane, leading to a fast and correct deinterleaving of the three emitters' pulses. HACOT-KS detects 0.01% outliers against 0.5% for IHACOT-KS.

5.1.2 . Robustness to outliers

The performances of the methods with varying proportions of outliers are evaluated. The proportion of outliers added to the signal varies between 0 and 90%. Although adding more than 50% of outliers in the signal does not make sense, the objective is to challenge the proposed methods in extreme situations. The performance criteria are averaged over 50 realizations to provide stable results. The outliers were generated uniformly between the range values for frequency and time of arrival. Alternatively, an exponential law generates pulse widths and level values because it favors the appearance of low values, making the outliers' values more realistic, as illustrated in Figure 5.7, which represents the previous signal in the (f_n, pw_n) and (g_n, toa_n) planes inf Figures 5.1d and 5.1c with 20% of outliers added.



(a) Method 1 output.

(b) Method 2 output.

Figure 5.3: Results of the HACOT-KS and IHACOT-KS developed in Sections 3.1 and 3.2 to deinterleave the signal in the Figure 5.1 based on Kolmogorov-Smirnov test for all grouping phases.

During the experiments, specifically during the first cluster grouping phase in IHACOT, following the 3D clustering, one observed that the results of the Kolmogorov-Smirnov and Student tests were relatively different. Therefore, IHACOT performance is evaluated by mixing the two tests: the Student test for the first grouping and the Kolmogorov-Smirnov test for the second grouping (identified as IHACOT-Mix) and the Kolmogorov-Smirnov test for the two groupings (identified as IHACOT-KS).

The previously mentioned metrics used to evaluate the algorithm's performances are presented in Figure 5.4. IHACOT is evaluated through two approaches: using only the Kolomogorov-Smirnov test for the cluster regrouping phases (IHACOT-KS) and by mixing the Student test for the first grouping and the Kolmogorov-Smirnov test for the second grouping (IHACOT-Mix). IHACOT-KS and IHACOT-Mix give higher results, as the ARI indicates. The better performance of these methods is due to the 3D-clustering, which allows us to deal with noisy data more efficiently by detecting outliers, leading to more reliable and stable results. The result is confirmed by the graph below: the number of pulses classified as outliers by HDBSCAN is higher for IHACOT-KS and IHACOT-Mix than HACOT-KS as the outliers added in the signal increase. Moreover, the final sets of pulses obtained from IHACOT-KS and IHACOT-Mix are composed of fewer outliers than those from HACOT-KS, as illustrated by the bottom right plot, meaning the 3-dimensional clustering provides less noisy sets of pulses. IHACOT-KS and IHACOT-MIX overestimate the number of emitters detected in the signal, as shown in the top-right plot. For our problem, the overestimated number of emitters detected is prioritized rather than underestimating them and having mixed pulses in a single cluster. IHACOT-KS and IHACOT-Mix seem more adapted to make deinterleaving.

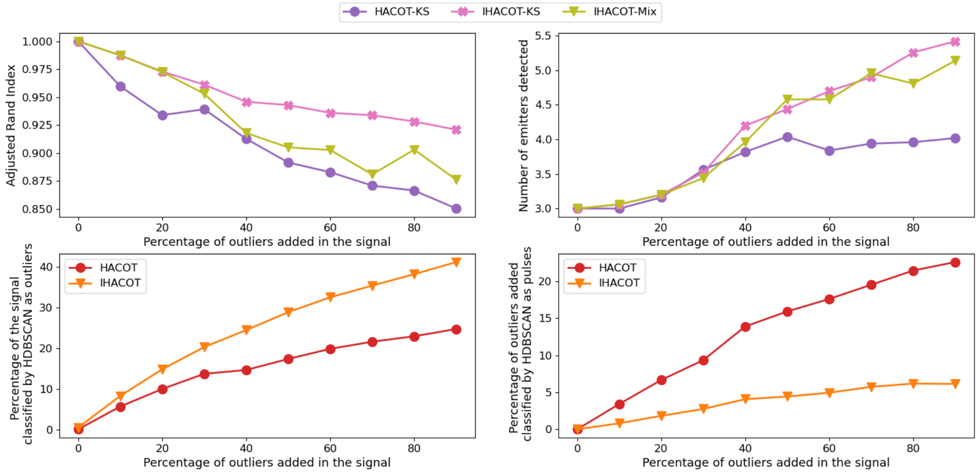


Figure 5.4: Performance of deinterleaving methods according to the outliers rate added with ARI, number of emitters detected, part of the signal classified by HDBSCAN as outliers, and part of outliers added in the signal classified in the sets of pulses. Each curve identifies a method.

Figures 5.7b, 5.7c and 5.7d show the results of HACOT and IHACOT applied on Figure 5.7a, each color represents a detected emitter, and -1 is the outliers class. In the figures, the results contrast HACOT and IHACOT; IHACOT-KS and IHACOT-Mix, based on a 3D clustering, deal better with adding outliers as 3D clustering detects more outliers. Note that some outliers corresponding to low-intensity pulses are grouped on both figures in class -1. Despite these outliers' presence, the algorithm identifies the 3 emitters present without mixing too much of their pulses between them. The 3D clustering separates the emitter's pulses and avoids mixing the pulses of several emitters in a single group.

5.1.3 . Sensitivity to estimation errors

Similarly to the previous section, the same process is applied to evaluate the performance of the algorithms by varying the noise level, and the metrics values are given in shown in Figure 5.5. The noise level in the estimated parameters of the PDW is varied by multiplying the baseline level given in Table 5.1 with a factor (*noise coefficient*) ranging from 1 to 10 as illustrated in Figure 5.8a. Likewise, 50 simulated signals per noise level were generated in the experimentation to average the results and obtain consistent results. An increasing noise level leads to the pulses spreading and overlapping the two emitters' pulses around 15.3 ns, making their pulses not differentiable.

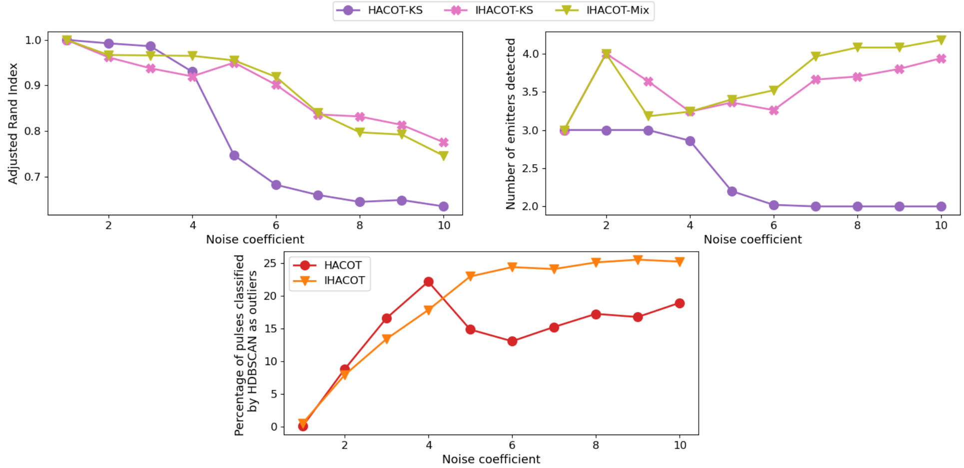


Figure 5.5: Performance of deinterleaving methods according to the noise added with ARI, number of emitters detected, and part of pulses classified by HDBSCAN as outliers. Each curve identifies a method.

Figures 5.8b, 5.8c, and 5.8d show the results of HACOT-KS, IHACOT-KS and IHACOT-Mix. Each color represents an emitter and (-1) the outliers. Conclusions similar to the experiment manipulating the outliers rate. HACOT-KS is insufficient to separate the pulses as the noise level increases. IHACOT separates the pulses better by applying 3D clustering, whereas HACOT fails to separate the pulses of the two emitters at around 15.3 ns. 3D clustering improves the pulse separation in the plane; IHACOT-KS and IHACOT-Mix overestimate the number of emitters detected after adding noise. These results are obtained following the regrouping phase of the clusters that do not perform enough aggregations. The pulses of the groups are superimposed, but the statistical tests lack robustness and significance in the presence of noise levels. An adjustment could be considered through new tests to correct this phenomenon and improve the results.

5.1.4 . Performance with missing pulses

This last part presents the deinterleaving results when pulses are missed during signal interception. Analogously to the previous parts, from the signal presented in Figure 5.1, a part of the pulses, varying from 0 to 90%, is randomly removed as shown in Figure 5.9a. The left plot represents the pulses in the (f_n, pw_n) plane according to different missing pulse rates. An increase in the missing pulse rate causes the pulses to spread in the plane. The right plot represents the PRI calculated from this reduced signal. The PRI values obtained increase as the missing pulse rate increases because the successive pulses do not belong to the same emitter.

The experiment results are averaged across 50 realizations for more robustness and presented in Figure 5.6. The results are stable, and the ARI is greater than 0.8, up to a missing pulse rate of 70%. The missing pulses allow them to spread in the plane and facilitate clustering.

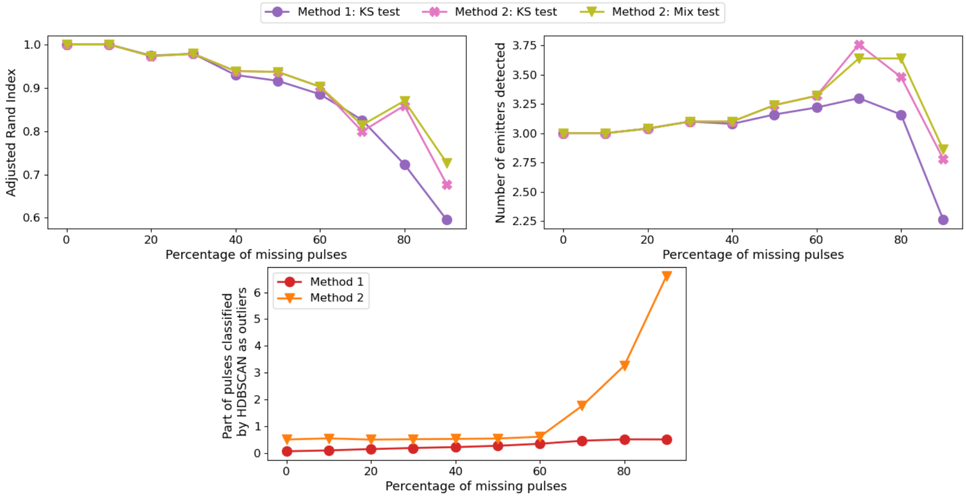
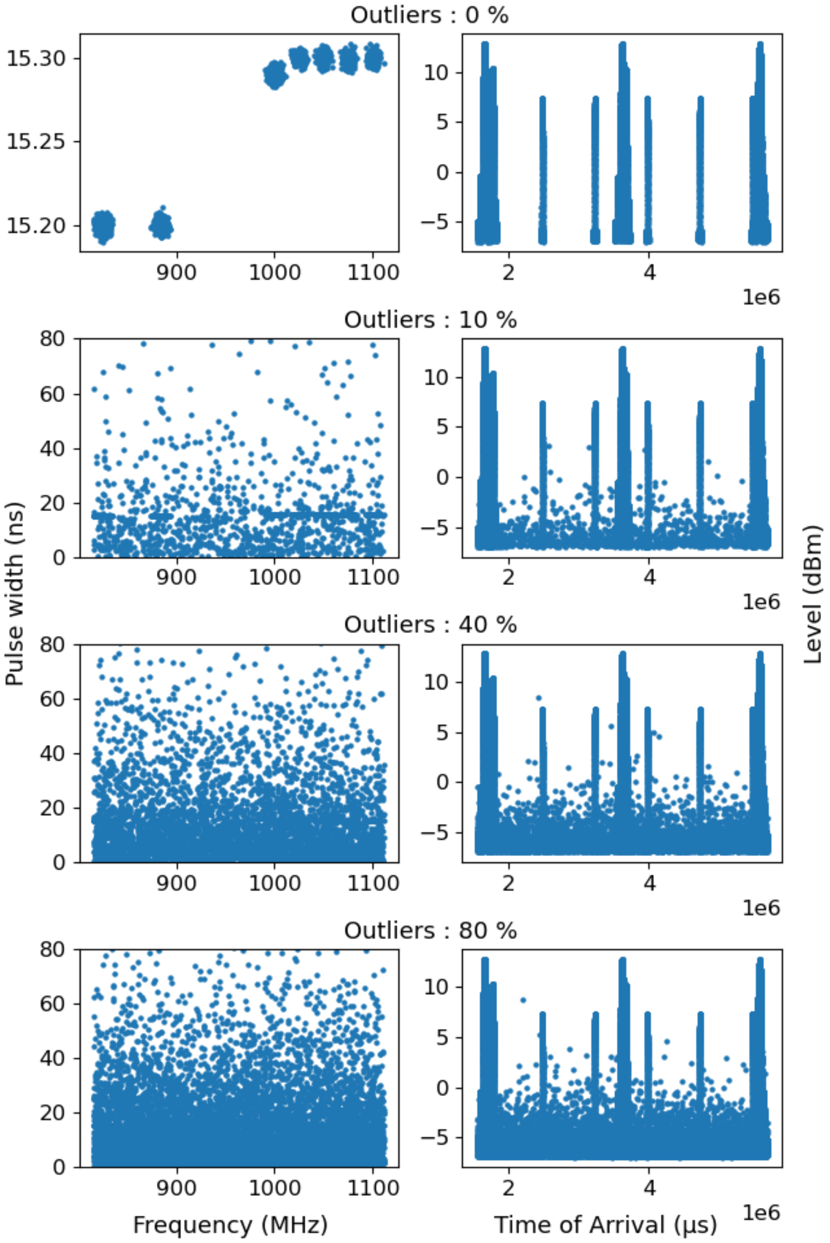


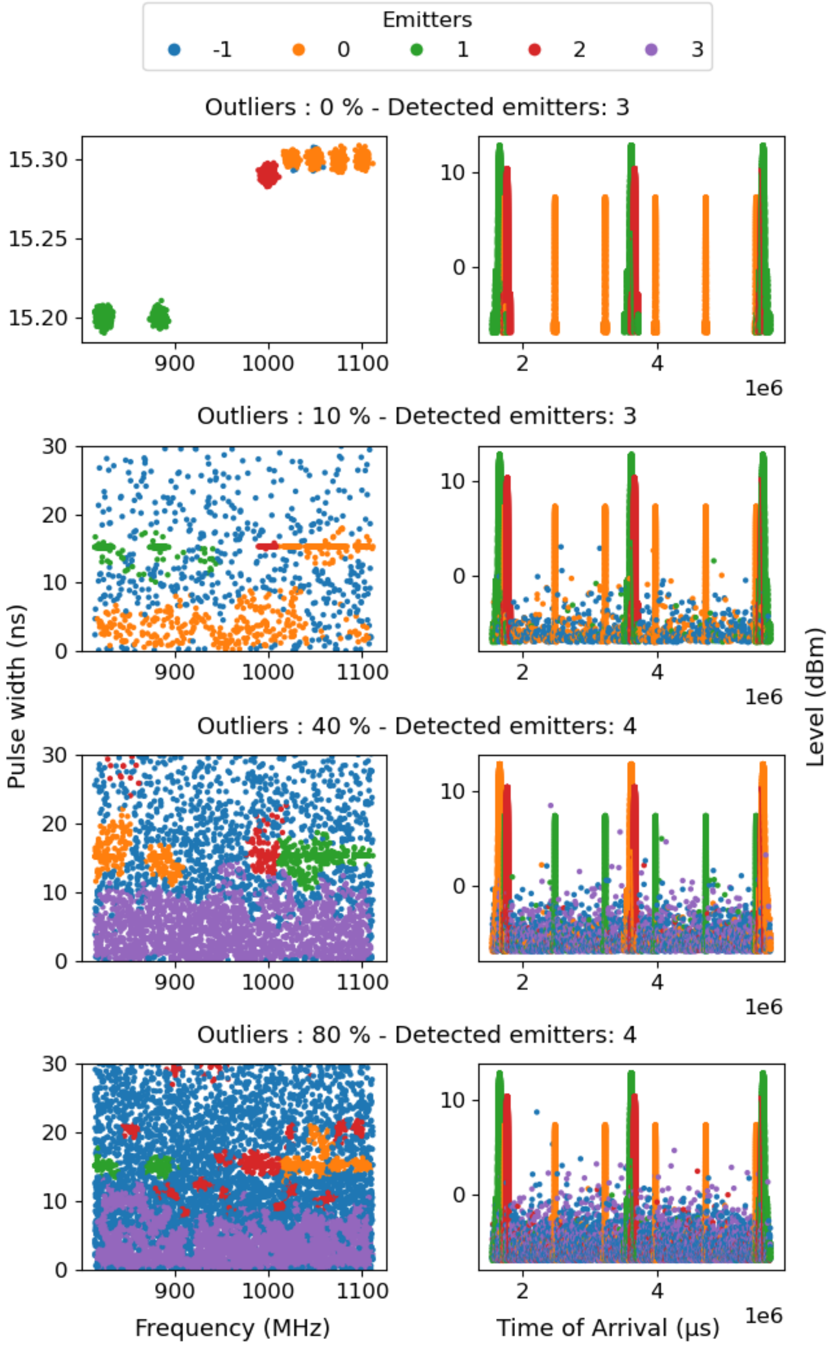
Figure 5.6: Performance of deinterleaving methods according to the missing pulse rate with ARI, number of emitters detected, and part of pulses classified by HDBSCAN as outliers. Each curve identifies a method.

Since the two deinterleaving methods are not based on using the PRI, they are not significantly impacted by the suppression of pulses, as shown in Figures 5.9b, 5.9c, and 5.9d. When the missing pulse rate increases, the number of emitters detected decreases because the clusters obtained by HDBSCAN do not contain enough information for the statistical test result to be significant.

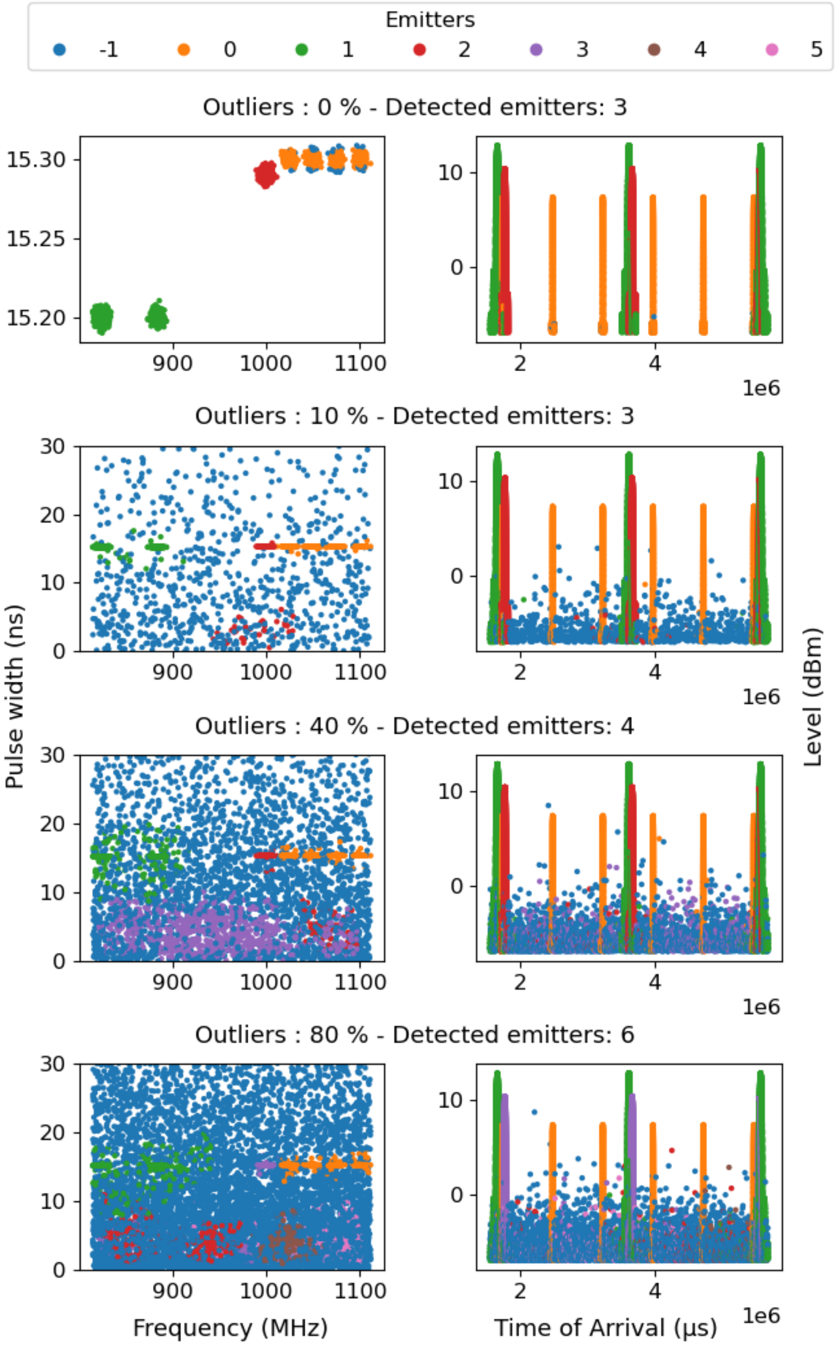
Additional quantitative results are available in [Les+23]. The authors reproduced the deinterleaving algorithm presented in Section 3.1. The HACOT algorithm was evaluated using various metrics, and the authors concluded that the hierarchical algorithm using optimal transport distances avoids merging clusters belonging to two different emitters, which is a critical condition for the context. Still, they pointed out that execution times could be long (see Section 3.1.6).



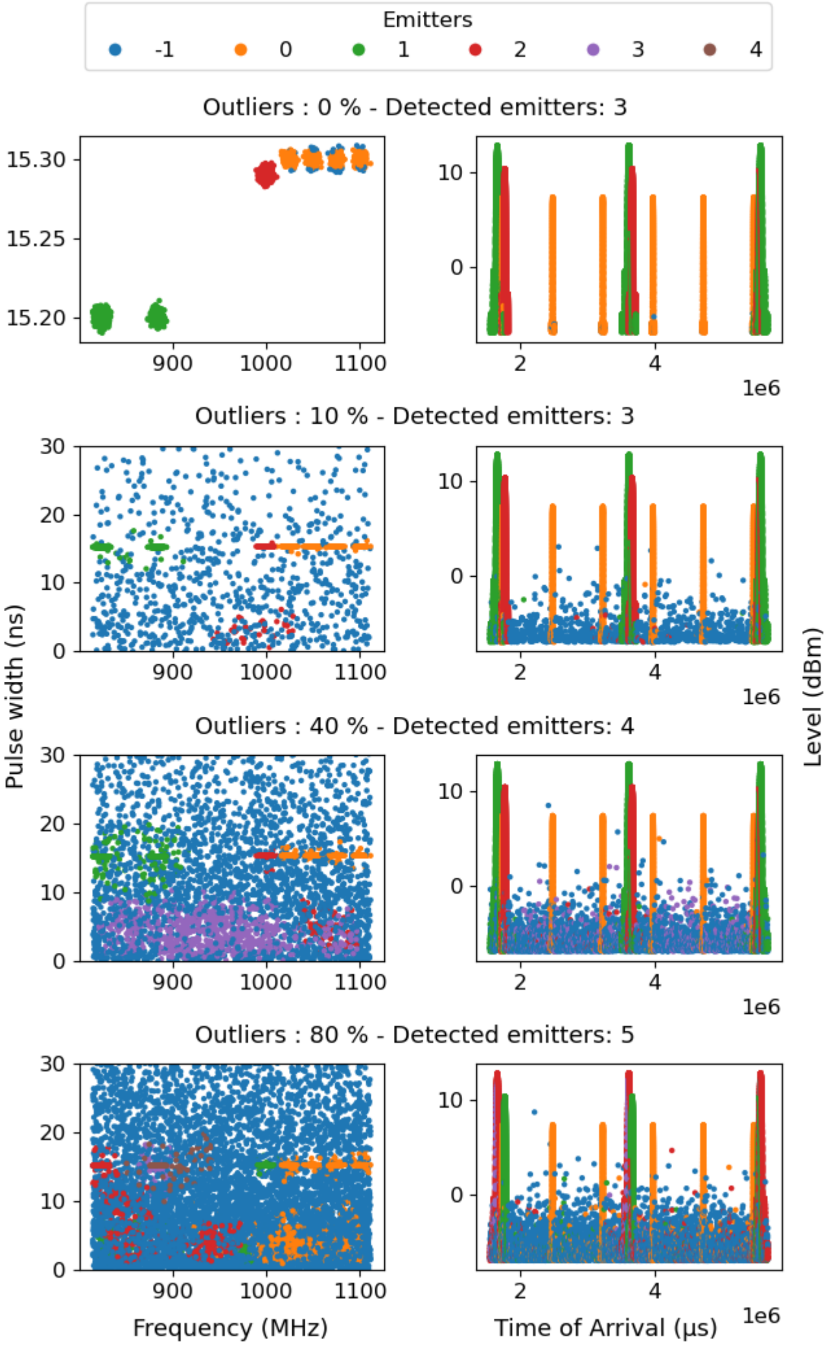
(a) Pulse’s representation.


Emitters
● -1
● 0
● 1
● 2
● 3
Outliers : 0 % - Detected emitters: 3
15.30
15.25
15.20
900
1000
1100
10
0
2
4
1e6
Outliers : 10 % - Detected emitters: 3
30
20
10
0
Outliers : 40 % - Detected emitters: 4
Outliers : 80 % - Detected emitters: 4
Pulse width (ns)
Level (dBm)
Frequency (MHz)
Time of Arrival (µs)

(b) HACOT-KS.

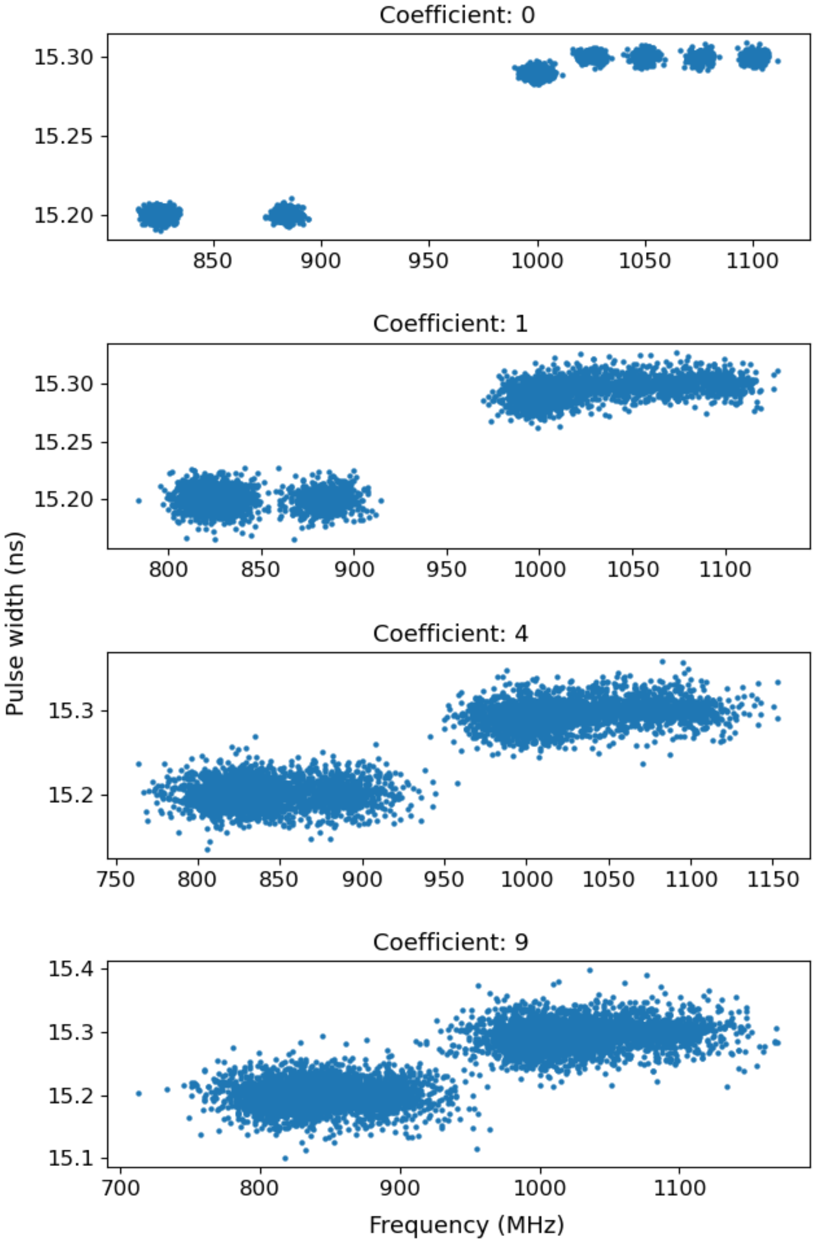

Emitters
● -1
● 0
● 1
● 2
● 3
● 4
● 5
Outliers : 0 % - Detected emitters: 3
15.30
15.25
15.20
900
1000
1100
10
0
2
4
1e6
Outliers : 10 % - Detected emitters: 3
30
20
10
0
Outliers : 40 % - Detected emitters: 4
Outliers : 80 % - Detected emitters: 6
Pulse width (ns)
Level (dBm)
Frequency (MHz)
Time of Arrival (µs)

(c) IHACOT-KS.

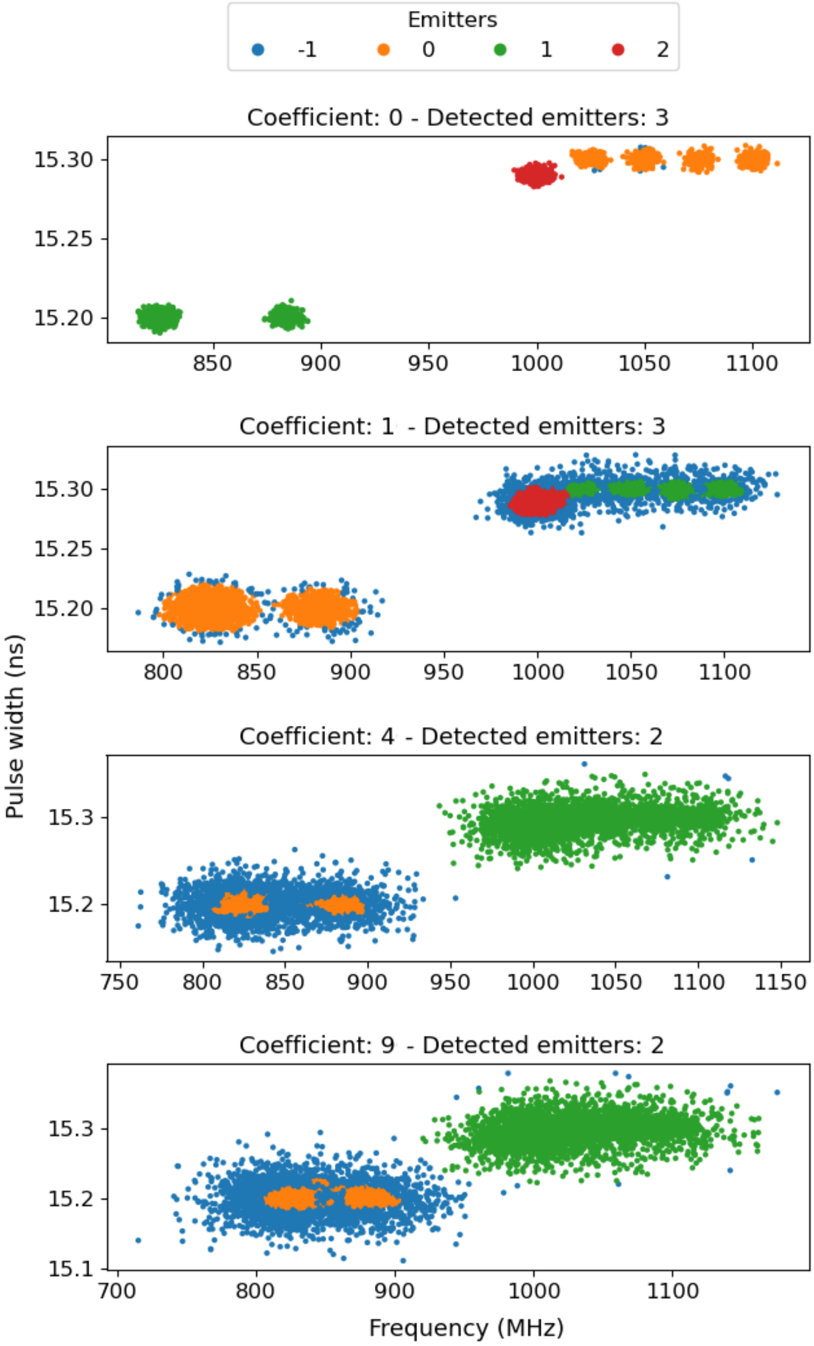


(d) IHACOT-Mix.

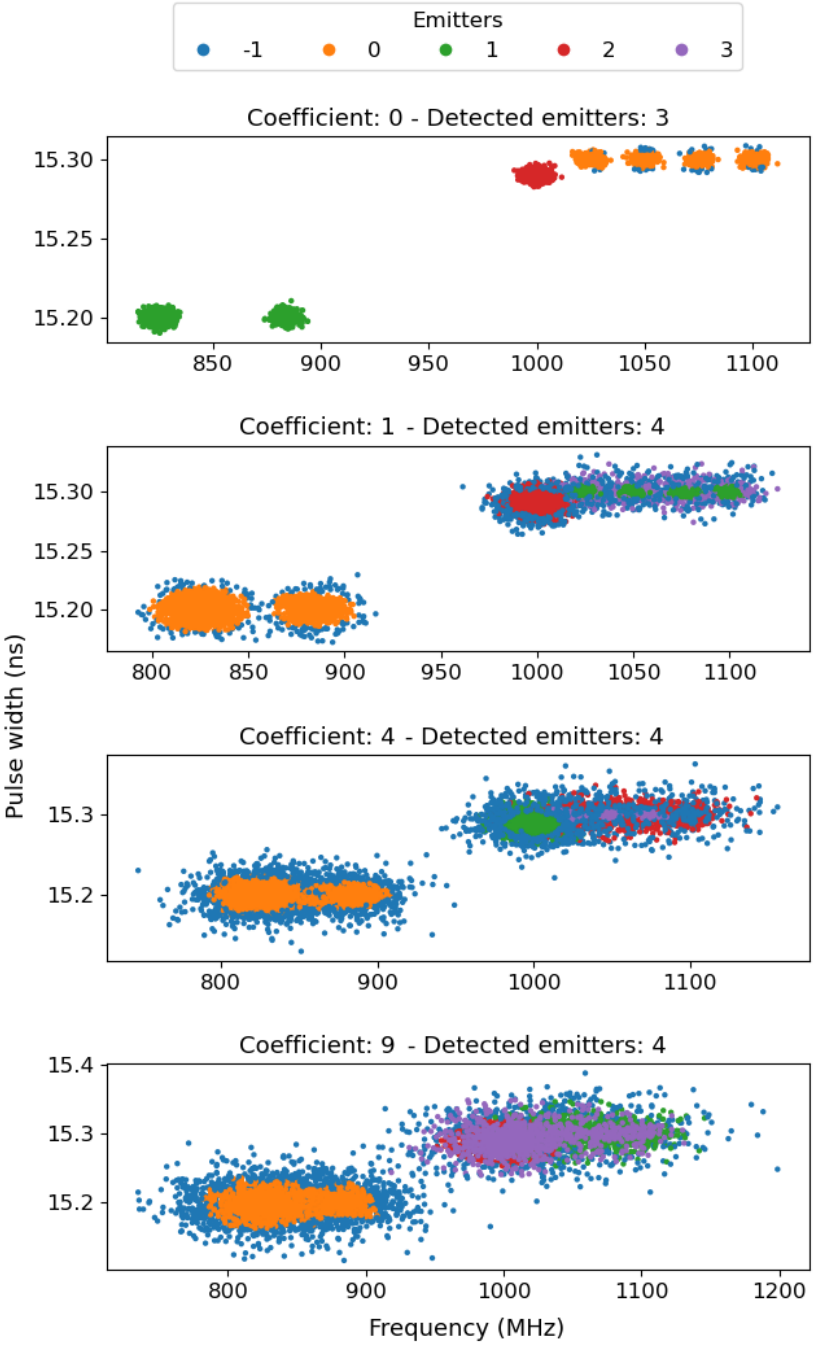
Figure 5.7: Results of HACOT and IHACOT applied with 0, 10, 40, and 80% of outliers added in the (f_n, pw_n) and (toa_n, g_n) plane. Each color represents a detected emitter, and -1 is the outliers class.



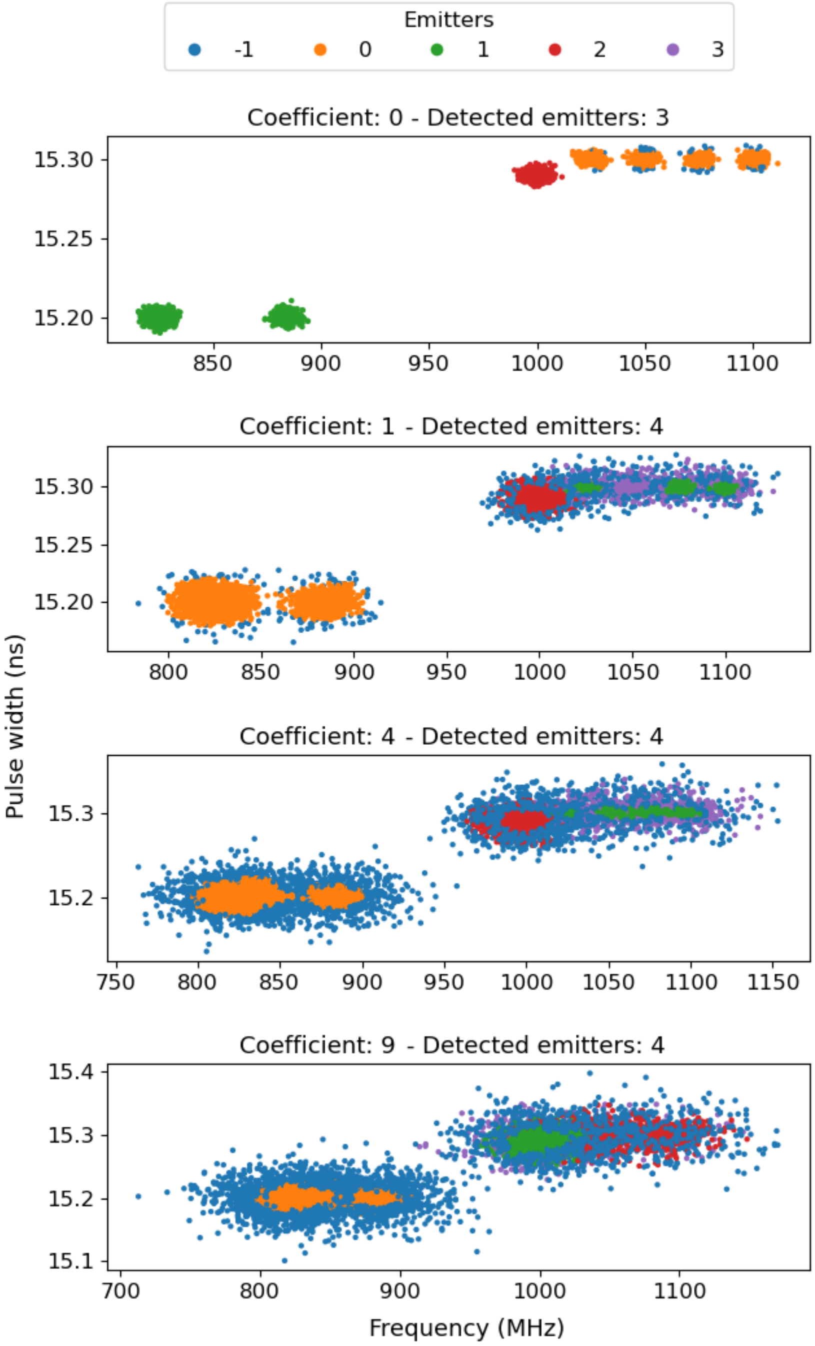
(a) Pulse's representation.


Emitters
-1
0
1
2
Coefficient: 0 - Detected emitters: 3
15.30
15.25
15.20
850
900
950
1000
1050
1100
Coefficient: 1 - Detected emitters: 3
15.30
15.25
15.20
800
850
900
950
1000
1050
1100
Coefficient: 4 - Detected emitters: 2
15.3
15.2
750
800
850
900
950
1000
1050
1100
1150
Coefficient: 9 - Detected emitters: 2
15.3
15.2
15.1
700
800
900
1000
1100
Pulse width (ns)
Frequency (MHz)

(b) HACOT-KS.

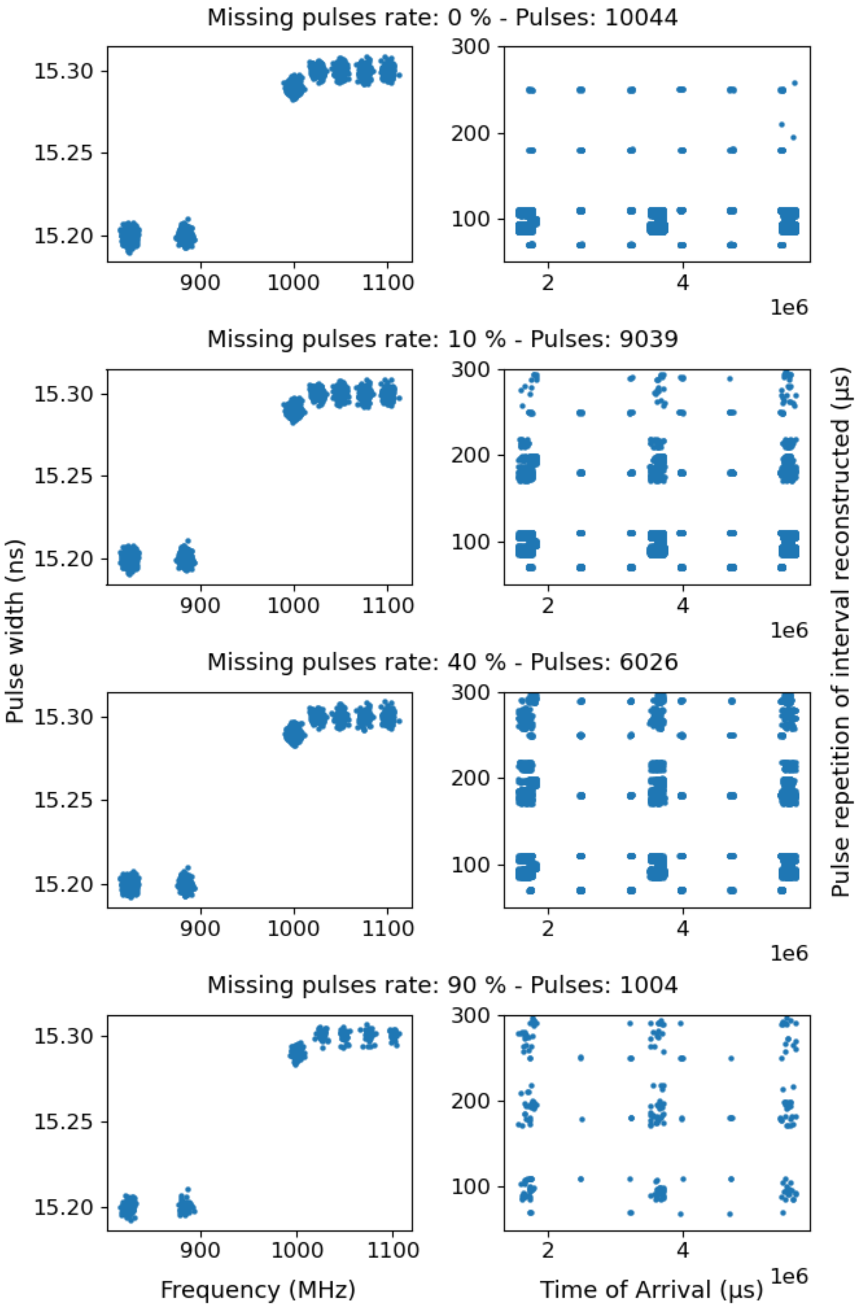

Emitters
-1
0
1
2
3
Coefficient: 0 - Detected emitters: 3
15.30
15.25
15.20
850
900
950
1000
1050
1100
Coefficient: 1 - Detected emitters: 4
15.30
15.25
15.20
800
850
900
950
1000
1050
1100
Pulse width (ns)
Coefficient: 4 - Detected emitters: 4
15.3
15.2
800
900
1000
1100
Coefficient: 9 - Detected emitters: 4
15.4
15.3
15.2
800
900
1000
1100
1200
Frequency (MHz)

(c) IHACOT-KS.

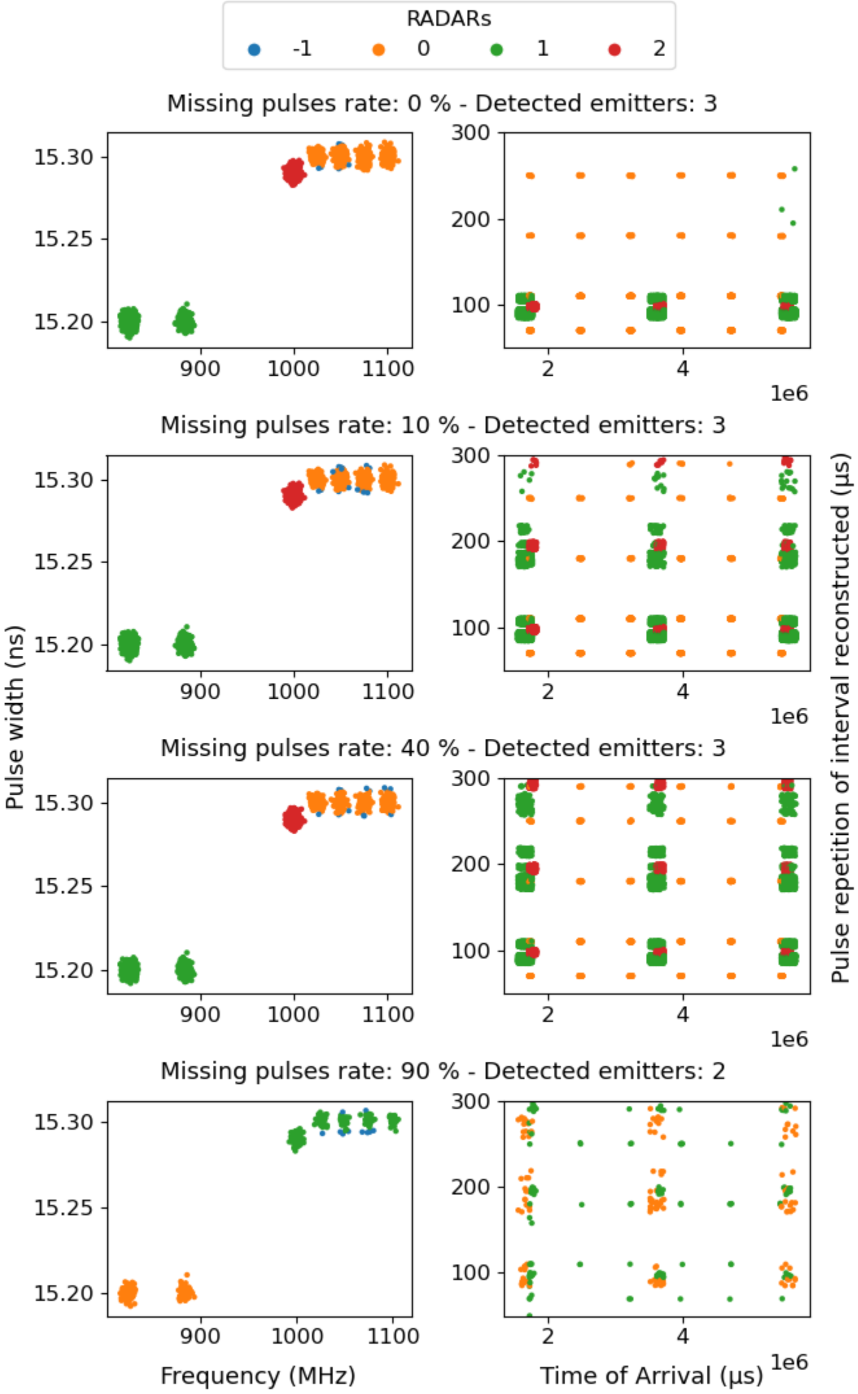


(d) IHACOT-Mix.

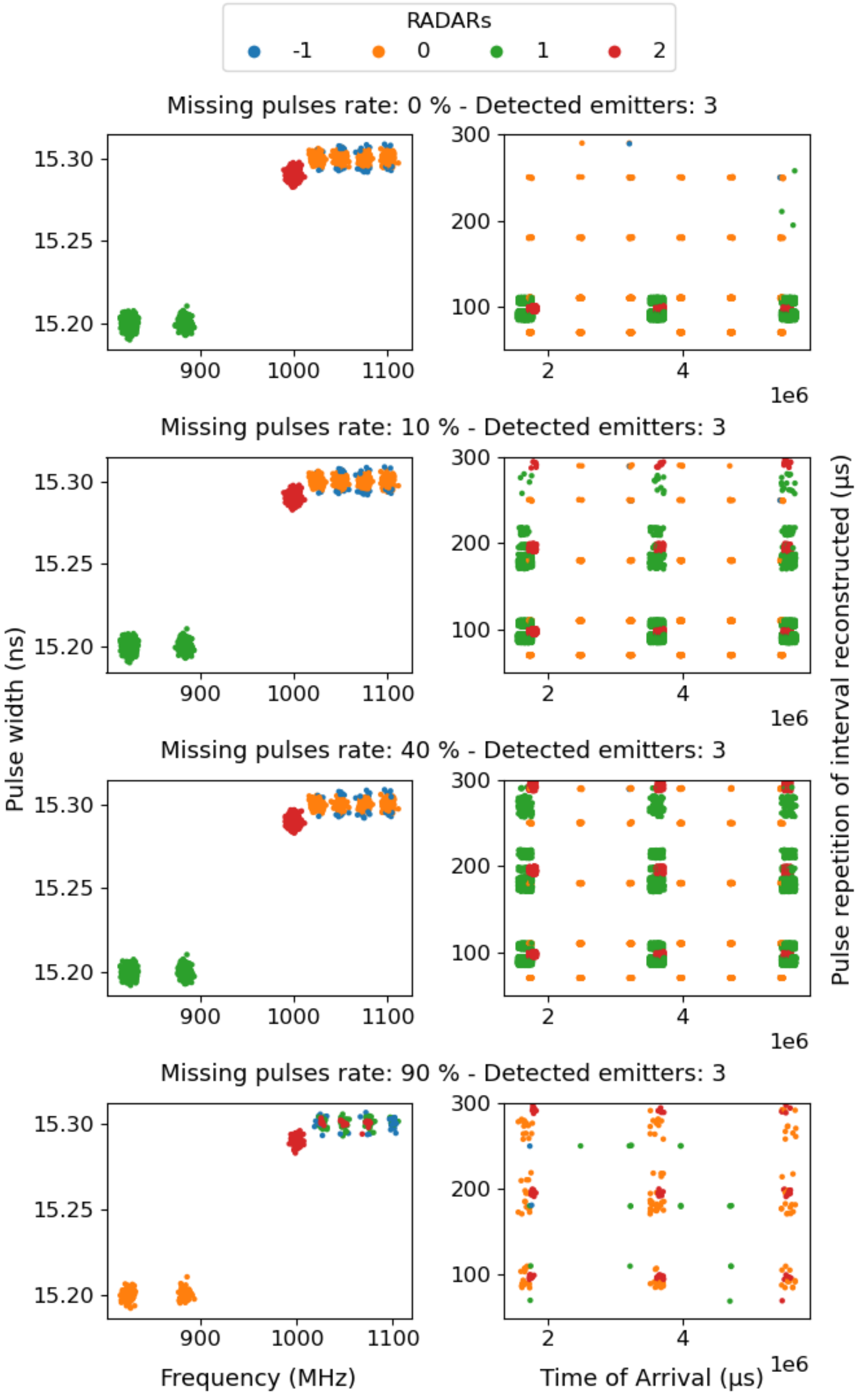
Figure 5.8: Pulse's representation in the (f_n, pw_n) plane by multiplying the baseline level with a factor ranging from 1 to 10 and their associated deinterleaving results.



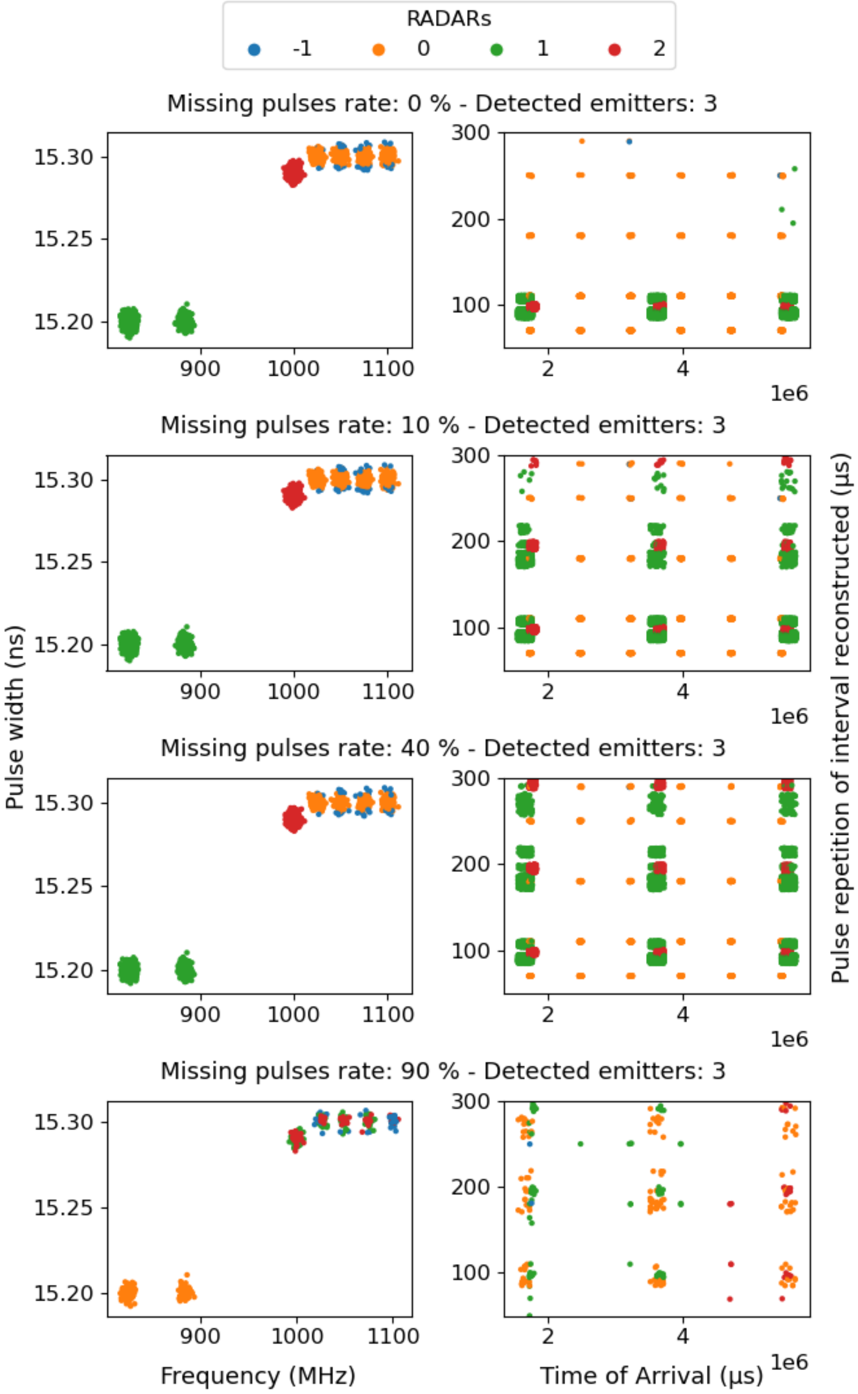
(a) Pulse's representation.


RADARs
● -1
● 0
● 1
● 2
Missing pulses rate: 0 % - Detected emitters: 3
Missing pulses rate: 10 % - Detected emitters: 3
Missing pulses rate: 40 % - Detected emitters: 3
Missing pulses rate: 90 % - Detected emitters: 2
Pulse width (ns)
15.30
15.25
15.20
900
1000
1100
Frequency (MHz)
Pulse repetition of interval reconstructed (µs)
300
200
100
2
4
1e6
Time of Arrival (µs)

(b) HACOT-KS.


RADARs
-1
0
1
2
Missing pulses rate: 0 % - Detected emitters: 3
Missing pulses rate: 10 % - Detected emitters: 3
Missing pulses rate: 40 % - Detected emitters: 3
Missing pulses rate: 90 % - Detected emitters: 3
15.30
15.25
15.20
900
1000
1100
300
200
100
2
4
1e6
Pulse width (ns)
Pulse repetition of interval reconstructed (μs)
Frequency (MHz)
Time of Arrival (μs)

(c) IHACOT-KS.



(d) IHACOT-Mix.

Figure 5.9: Pulse's representation in the (f_n, pw_n) plane randomly removing pulses with a constant ranging from 1 to 9 and their associated deinterleaving results.

5.2 . Evaluation of identification results

Section 4.1.2 proposes a new representation of the pulse repetition interval distribution to consider the missing pulse rates. The section evaluates the identification method by manipulating four parameters (adding outliers, adding noise, missing pulses, and mixing the pulses) as deinterleaving algorithms in Section 5.1. The algorithm's efficacy is evaluated by analyzing the probability of correctly identifying the correct emitter in the algorithm outputs.

As previously explained in Section 2.5, deinterlaving and identification are generally done simultaneously. Nevertheless, the two steps are considered independent in this manuscript. Comparisons with other methods have not been made because existing methods are very different from those presented here; the existing methods are mainly based on extracting waveform modulation or PRI type among a shortlist, while the methodology can identify up to the RADAR sub-operation mode.

A new emitter database presented in Table 5.2, deliberately grouping emitters with very similar characteristics (identical frequencies and pulse width, similar PRIs, PRIs overlapping, etc.) was constructed to highlight the robustness of the identification algorithm. The representation of the PRI proposed in Section 4.1.2 was applied to this database, considering a missing pulse rate up to 60%. For the following, the class C_{40} will designate the modeled class of the Emitter C in our identification database with the PRI modeled according to a missing pulse rate of 40% in the set of pulses.

Simulated signal gathering four sets of pulses (330 pulses of Emitter A, 971 pulses of Emitter B, 858 pulses of Emitter C, and 1126 pulses of Emitter D), identifying by color is displayed in Figure 5.10 based on Table 5.2. The sets of pulses deliberately contain few pulses and heterogeneous size to highlight the method's robustness as a probability distribution represents them.

Name	**Frequency (MHz)**	**Pulse width (ns)**	**PRI (μs)**
A	1025, 1050, 1075 1100, 1125, 1150	2	1520, 1540, 1560, 1580 1600, 1620, 1640, 1660 1680, 1700, 1720, 1740 1760, 1780, 1800, 1820 1840, 1860, 1880, 1900 1920, 1940, 1960, 1980
B	972	1.2	2000, 2200, 2400
C	972, 1072, 1112	4	2200
D	1000	4	1700, 2200
E	825, 884	2.9	1700, 1720, 1740, 1760 1780, 1800, 1820, 1840 1860, 1880, 2080, 2120 2160, 2200
F	800	3.7	1700, 1800, 1900, 2000 2100, 2200, 2300, 2400
g	938	2.3	Min: 1700, Max: 1900

Table 5.2: Simulated emitters characteristics.

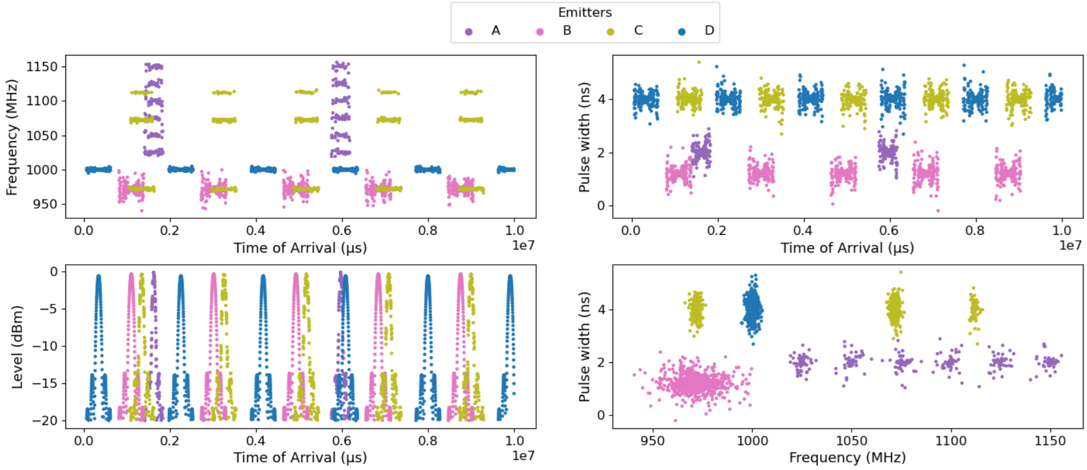
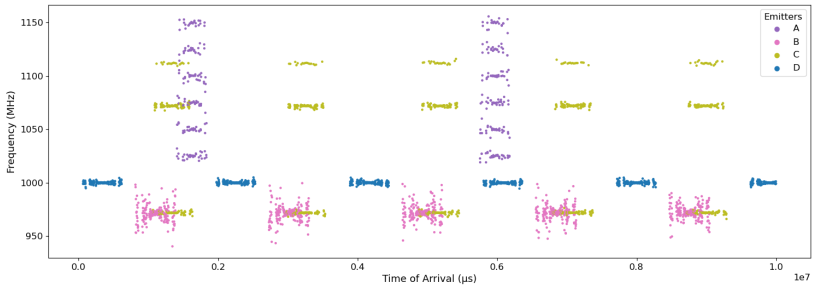
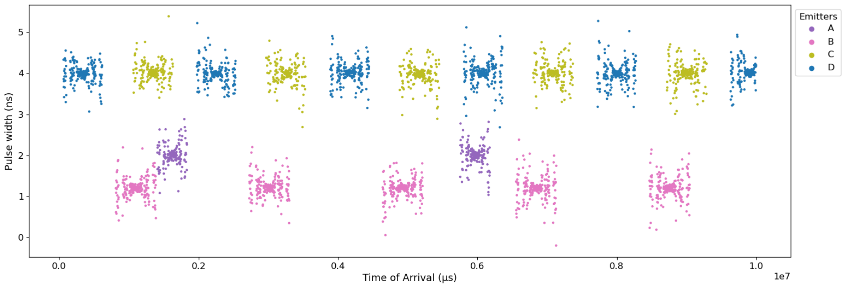


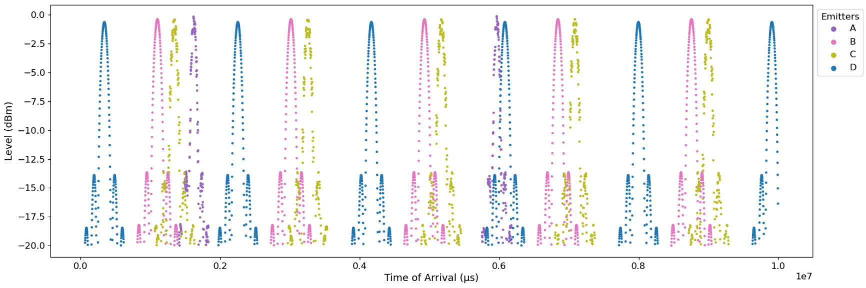
Figure 5.10: Simulated signal gathering 330 pulses of Emitter A, 971 pulses of Emitter B, 858 pulses of Emitter C, and 1126 pulses of Emitter D, identifying by a color.



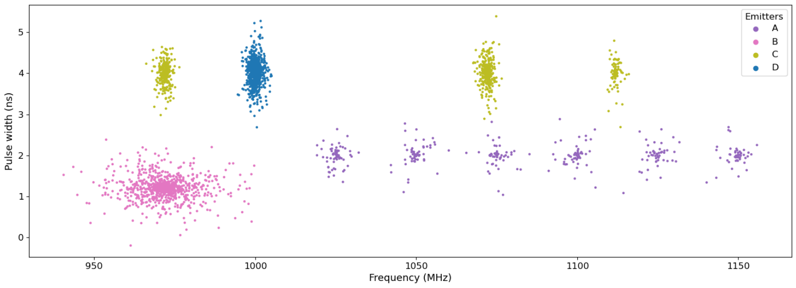
(a) Frequency x time of arrival plane.



(b) Pulse width x time of arrival plane.



(c) Level x time of arrival plane.



(d) Frequency x Pulse width plane.

Figure 5.11: Simulated signal gathering 330 pulses of Emitter A, 971 pulses of Emitter B, 858 pulses of Emitter C, and 1126 pulses of Emitter D, identifying by a color.

5.2.1 . Performance with missing pulses

The identification method is first evaluated and analyzed when not all pulses from an emitter are present. A part of the pulses, varying from 0 to 60%, is randomly removed from each set of pulses as highlighted in Figure 5.13. The left plot represents the pulses in the (f_n, pw_n) plane according to different missing pulse rates. The differentiated time of arrival $(dtoa_n)$ is computed on the right plot. An increase in the missing pulse rates causes the $dtoa_n$ to spread in the plane. Then, the identification algorithm is applied to each set of pulses, and the performance criteria are averaged over 50 realizations to provide stable results.

The probability of correctly identifying the correct emitter in each pulse set is presented in Figure 5.12. Regardless of the missing pulse rates, the algorithm can correctly identify the emitter in each set of pulses, even when more than 60% of the pulses are missing. The PRI modeling approach proves highly effective by incorporating new PRI values due to missing pulses, even when emitters share similar characteristics, enhancing the methodology's robustness.

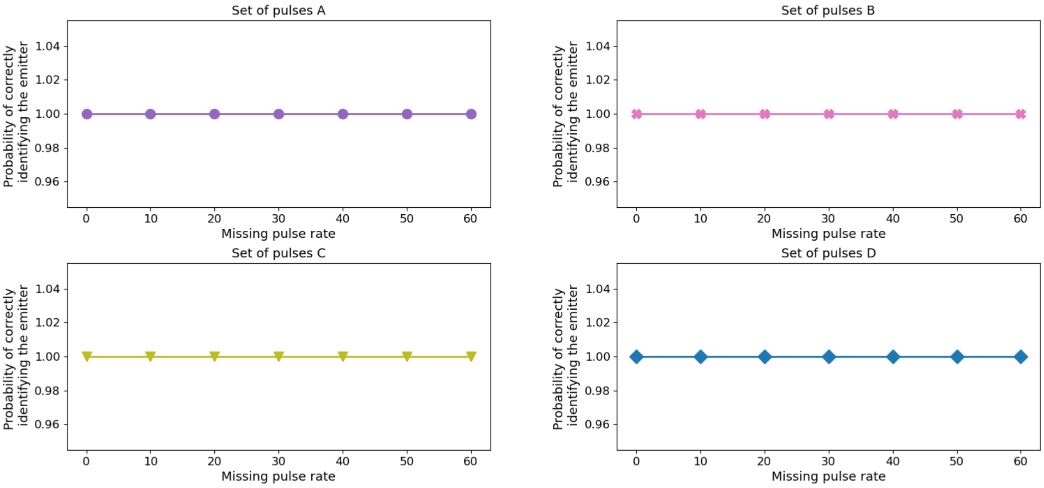


Figure 5.12: Probability of identifying the correct emitter in each set of pulses according to several missing pulse rates.

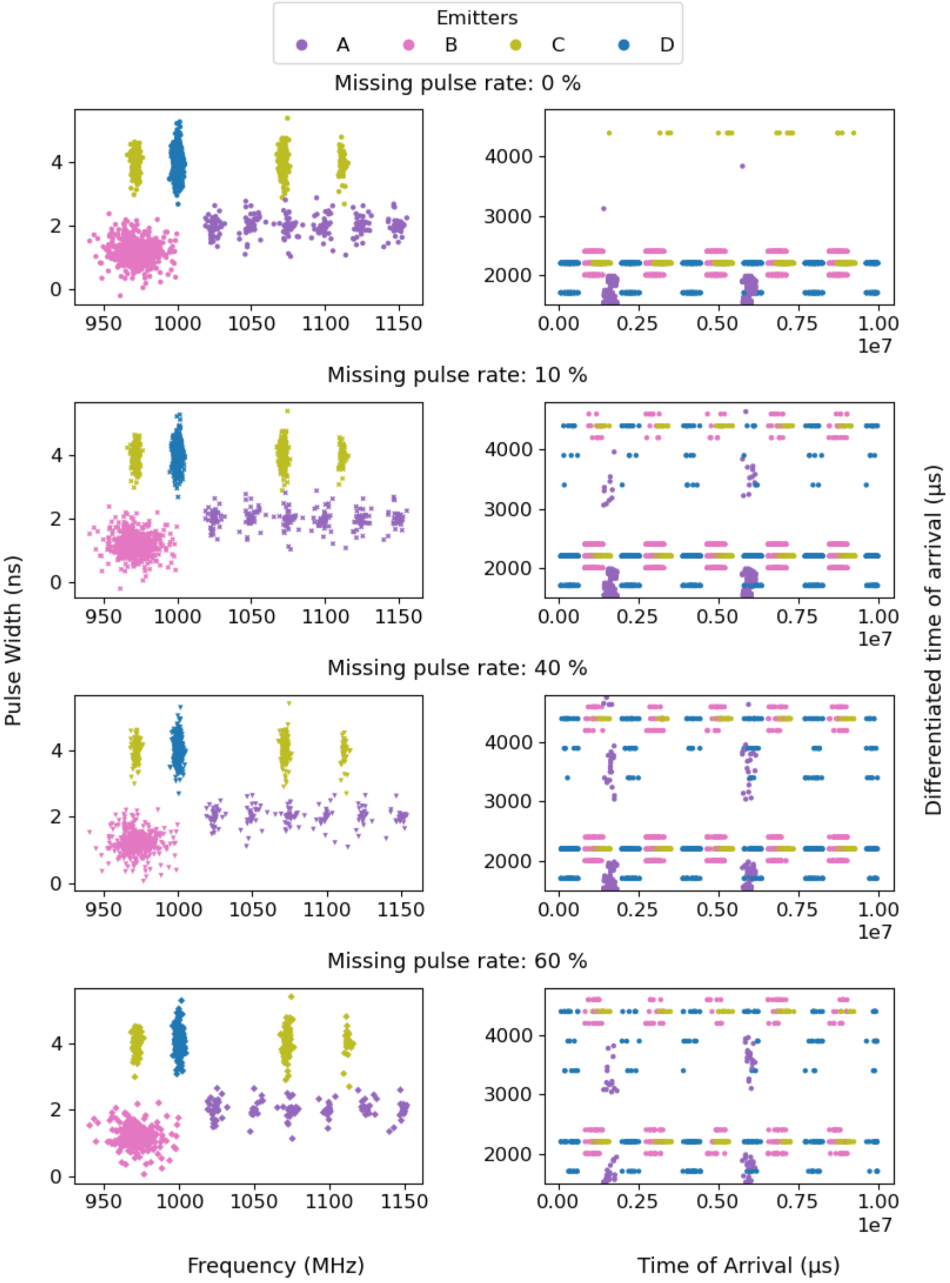


Figure 5.13: Pulse's representation in the (pw_n, toa_n) and $(dtoa_n, toa_n)$ planes according to different missing pulse rates.

5.2.2 . Robustness to outliers

Similarly to Section 5.1.2, the methods' performances with varying proportions of outliers are evaluated. The proportion of outliers added to the signal varies between 0 and 50%. The performance criteria are again averaged over 50 realizations to provide stable results. The outliers were generated uniformly between the range values for frequency and time of arrival for each set of pulses as illustrated in Figure 5.16, which represents the previous signal in the (f_n, pw_n) and $(dtoa_n, toa_n)$ planes with multiple percentages of outliers added.

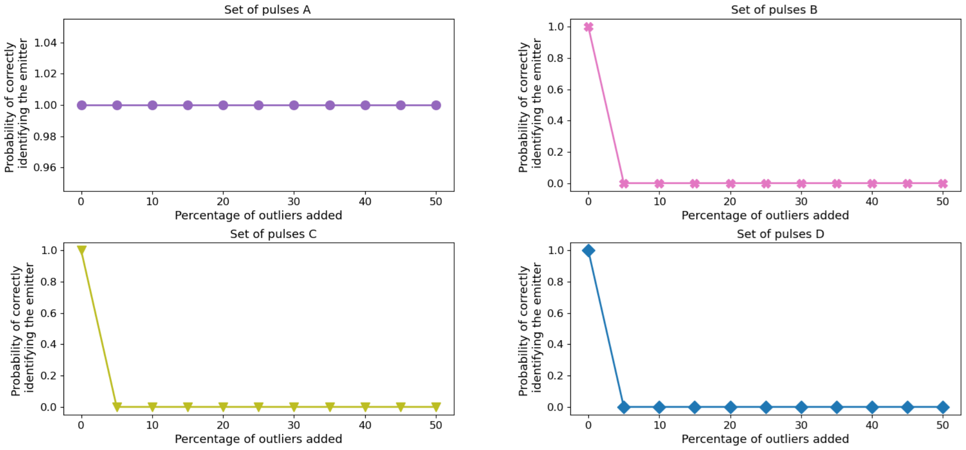
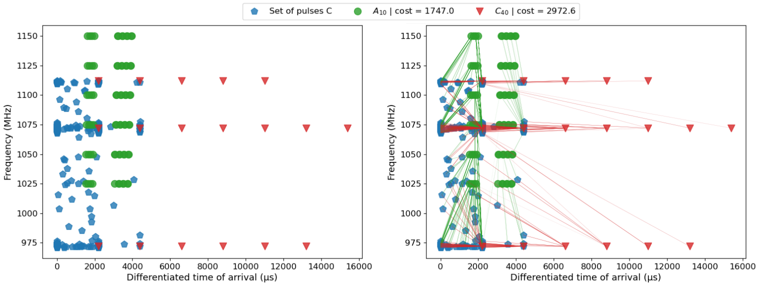


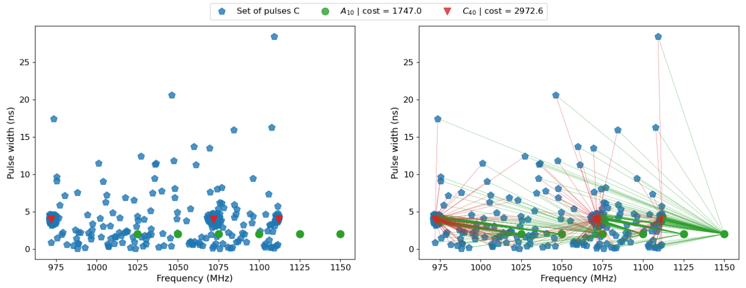
Figure 5.14: Performance of identification methods according to the outliers added by analyzing the probability of identifying the correct emitter for each set of pulses.

Adding outliers distorts the sets of pulses distributions and ultimately corrupts the identification for sets of pulses B, C, and D, as shown in Figure 5.14, due to their similarity presented in the Table 5.2. Frequency is a highly discriminating characteristic of an emitter; Emitters characterized by a limited number of frequencies are susceptible to rapidly being affected by distortions in their frequency distribution when outliers are introduced, compromising the identification results. In contrast, the Set of pulses A, characterized by five frequencies, can assimilate added outliers with minimal impact, rendering them practically negligible in the identification process. We note that the algorithm will often identify the emitter present in the set of pulses as the one having average characteristics compared to those of the other classes in the plane.

Figures 5.15 overlays the simulated pulses from Emitter C with 40% of outliers added on the left plot and the characteristics points of the first output returned by the algorithm (A_{10}) and the characteristics points of true modeled class (C_{40}) in the set of pulses from the identification database and the right plots displayed the transport planes. The characteristic points of class A_{10} do not superimpose very well on the data, but they remain relatively close. Adding outliers spreads the pulses in the plane, so the three frequencies of Emitter C are not easily distinguishable, and other pri values appear. The algorithm will identify the emitter present in the signal as the one with average characteristics compared to those of the other classes, that is to say, the Emitter A, because it has frequencies distributed throughout the plane, making moving pulses and outliers to this frequency less expensive. In the hierarchical methods developed for deinterleaving, clusters were represented as histograms; binarization can significantly influence identification results, especially if the modeled PRI is composed of small values. Too fine Bins could distort the distribution in the presence of many outliers, while too-wide bins might not correctly capture the data trend. This is all the more true when looking at Figure 5.15b; adding outliers spreads the pulse width values even further. The use of histograms does not seem suitable when there are many outliers in a signal because it is difficult to find a good compromise to fix the bin numbers. As specified in Section 1.1, the optimal transport implies a displacement of all points. It might be interesting to use an unbalanced optimal transport that would be more flexible, would not displace all the points, and would be more robust to outliers.



(a) Frequency and Differentiate time of arrival plane.



(b) Frequency and Pulse width plane.

Figure 5.15: Classification results when 40% of outliers is added for Set of pulses C.

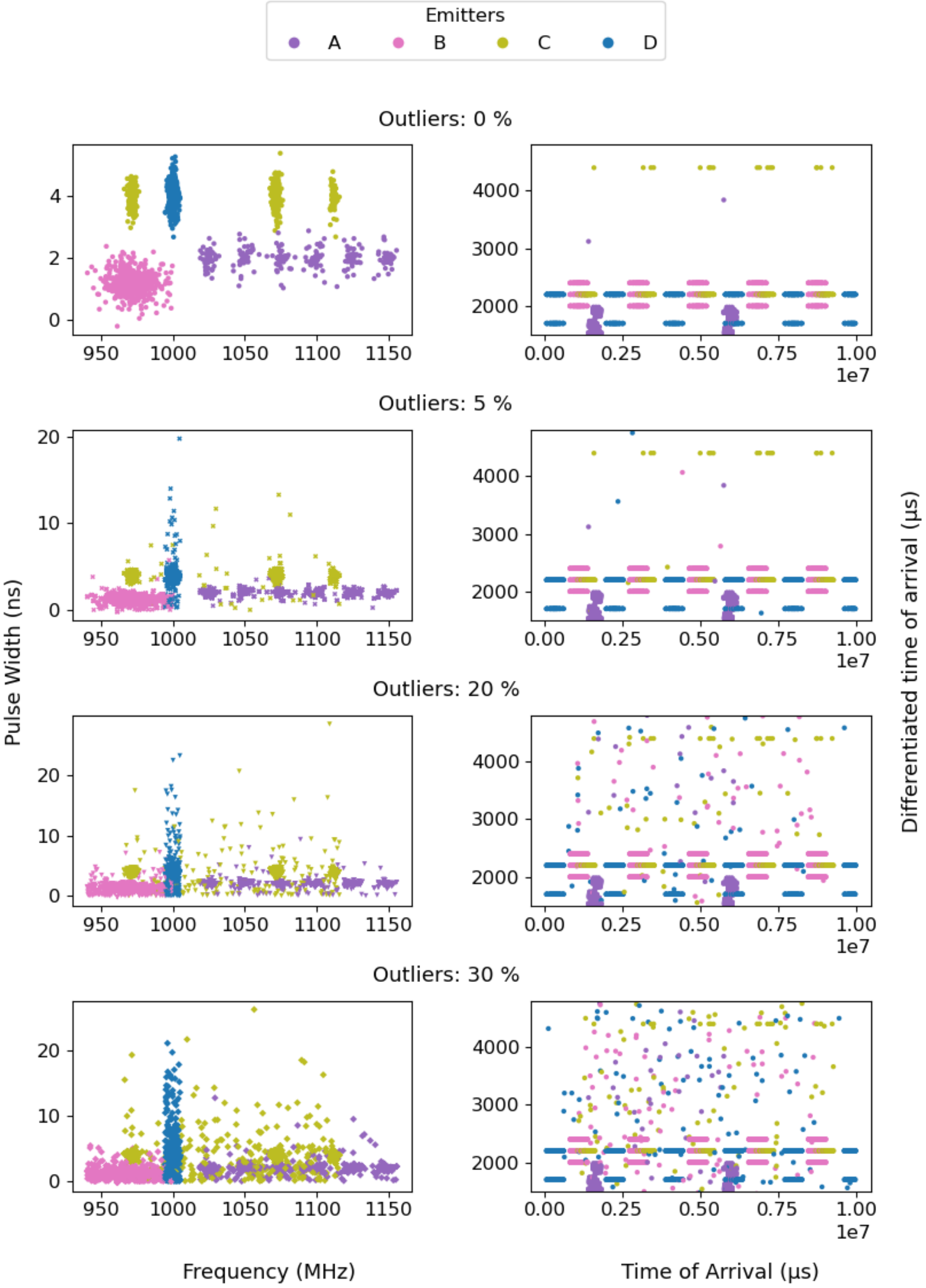


Figure 5.16: Pulse's representation in the (pw_n, toa_n) and $(dtoa_n, toa_n)$ planes according to different outliers rates added.

5.2.3 . Sensitivity to estimation errors

Similarly to the previous section, the same process is applied to evaluate the performance of the identification algorithm by varying the noise level. The noise level in the estimated parameters of the PDW is varied by multiplying the sets of pulse levels with a factor (*noise coefficient*) ranging from 1 to 90 as illustrated in Figure 5.19. Likewise, 50 simulated signals per noise level were generated in the experimentation to average the results and obtain consistent results. An increasing noise level leads to the pulses spreading for each set.

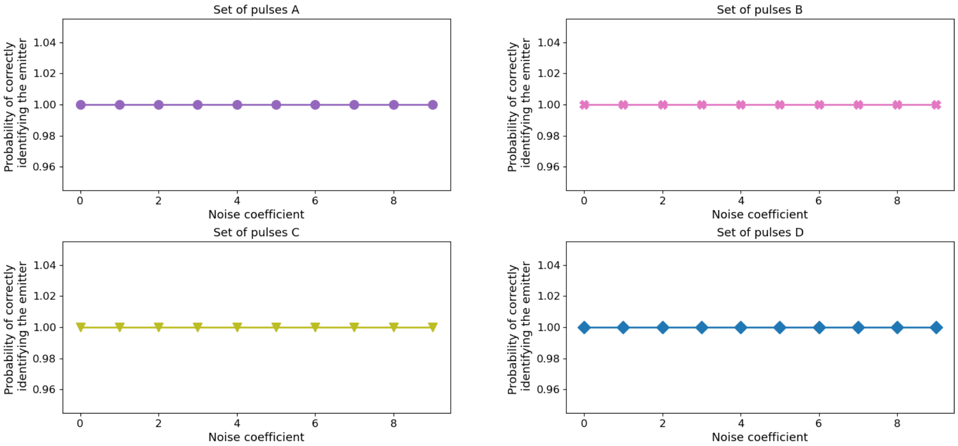
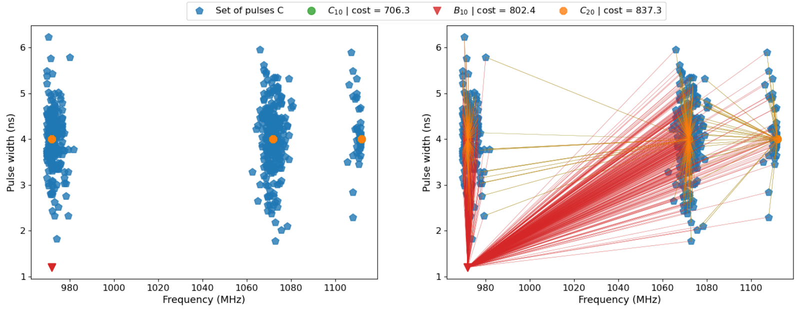
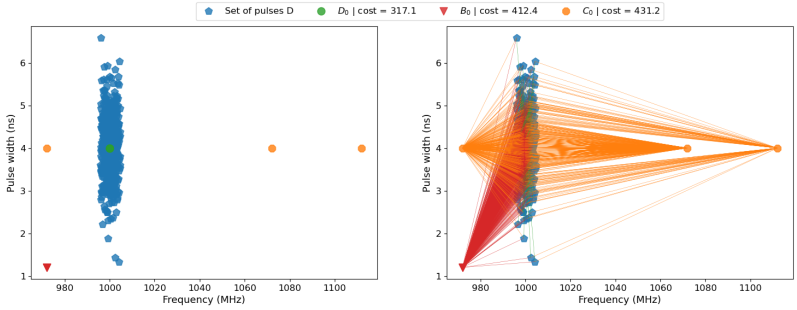


Figure 5.17: Performance of identification algorithm by analyzing the probability of identifying the correct emitter for each set of pulses when the variance level of the set of pulses is multiplied by a coefficient.

Regardless of the noise level of the pulse added, as shown in Figure 5.19, the algorithm can identify the emitter present. These results can be explained by the added noise level only impacting the pulses spreading across the planes. The computation of $dtoa$ is not impacted and remains just as discriminating, allowing the integration of the frequency and pulse width measurement errors. This spread is smoothed by representing the data in histograms, as illustrated in Figures 5.18. The figures on the left respectively represent the points of the sets of pulses C and D when the baseline has been multiplied by forty and the characteristic points of the three closest classes identified by the algorithm. The right plots show the transport planes.



(a) Set of pulses C outputs.



(b) Set of pulses D outputs.

Figure 5.18: Classification results when the baseline noise level is multiplied by 4 for Set of pulses C & D in the (f_n, pw_n) plane.

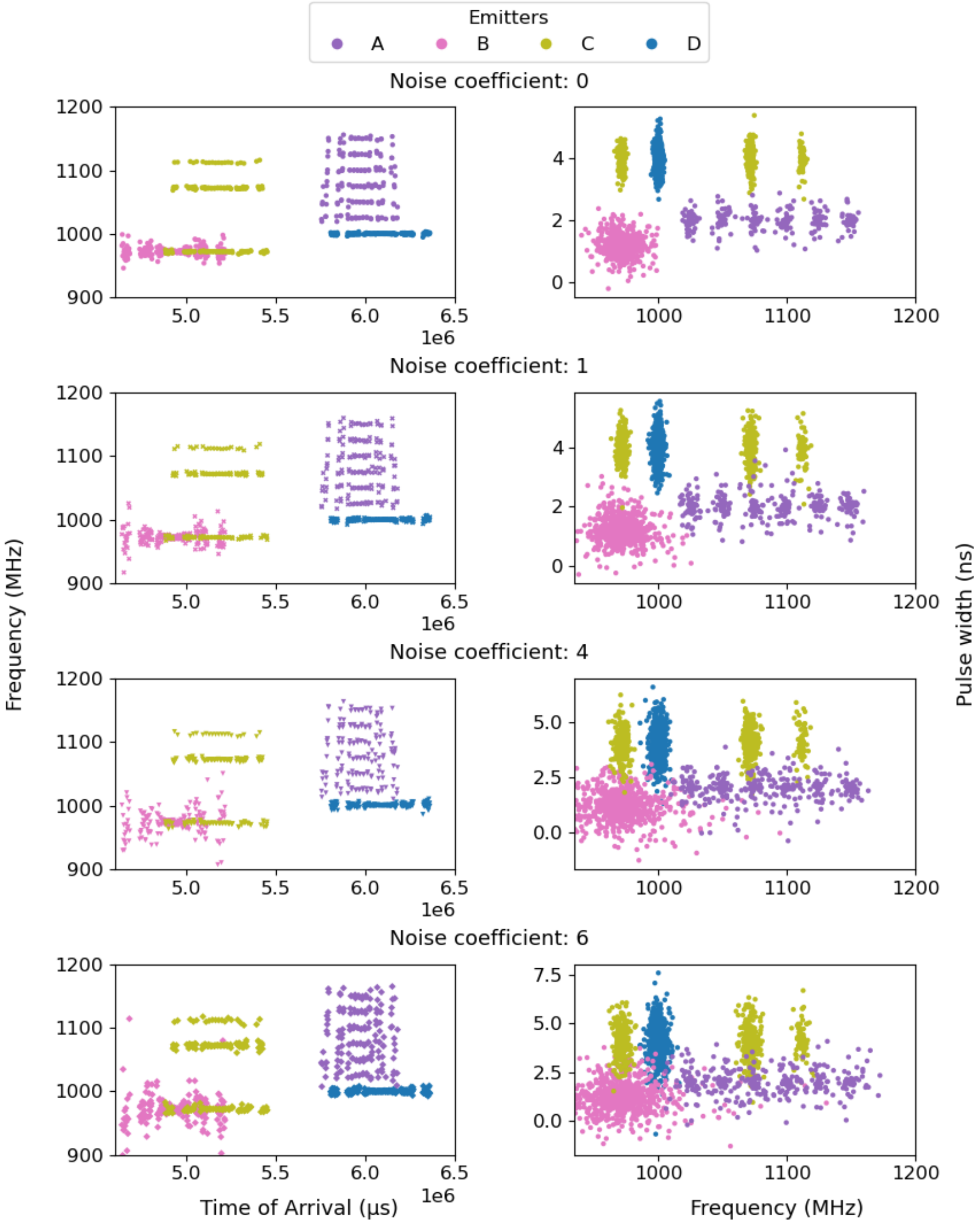


Figure 5.19: Pulse's representation in the (f_n, toa_n) and (pw_n, f_n) planes when the baseline noise level is multiplying by a coefficient for each set of pulses.

5.2.4 . Effectiveness against mixing pulses

The last experiment evaluates the robustness of the identification algorithm when a set of pulses does not contain pulses from a single emitter. The identification is made from sets of deinterleaving pulses, assuming that the sets of pulses contain the pulses of a single RADAR. However, the deinterleaving process may fail to separate the pulses correctly, for example, when the signal is too noisy, poorly estimated, or the emitters have too similar characteristics. A percentage of a randomly selected pulse from another set ranging from 0 to 80 is replaced by a given pulse set, and then the identification algorithm is applied. The results are again averaged over 50 realizations.

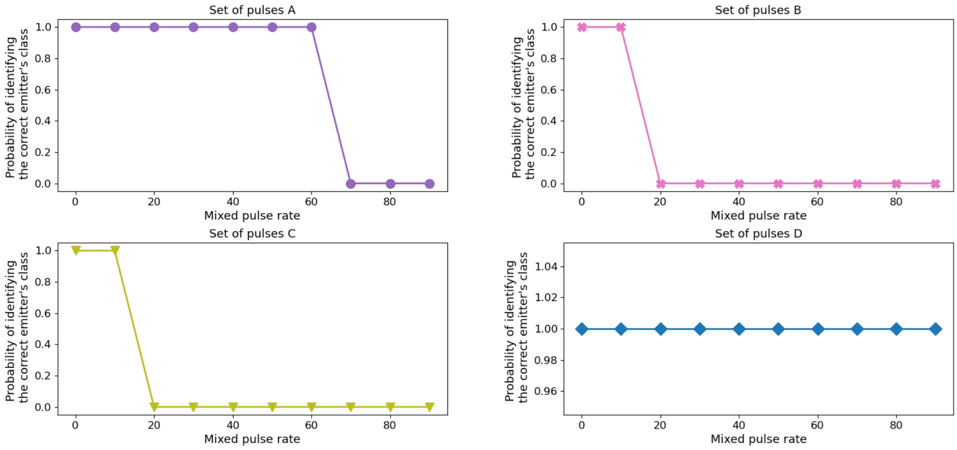
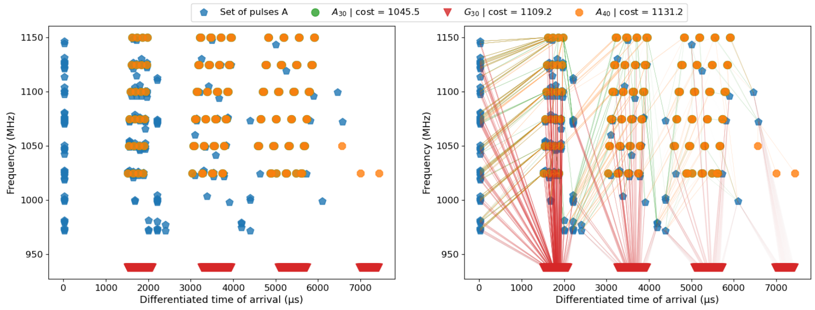


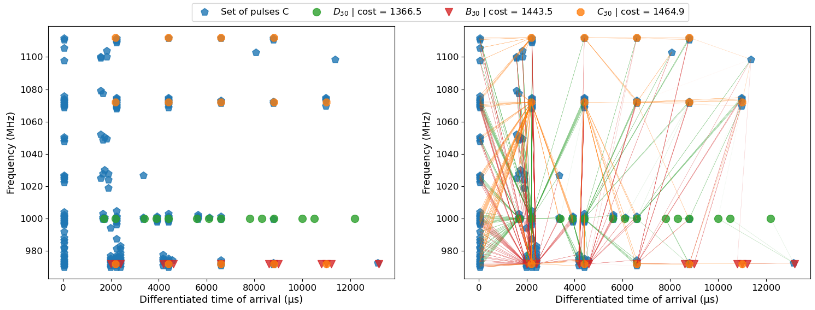
Figure 5.20: Performance of identification methods by analyzing the probability of identifying the correct emitter for each set of pulses with mixing pulses.

Figures 5.22 represent the set of pulses A in the (f_n, pw_n) and $(dtoa, toa)$ planes when the mixing rate increases. When the mixing rate increases, the sets of pulses contain pulses belonging to other emitters; therefore, the $dtoa$ is calculated from successive pulses not belonging to the same emitters, explaining its spread on the graphs on the right. High values of $dtoa$ will distort the distributions to be compared with the classes of emitters in the classification database as shown in Figures 5.21. The pulse sets A and D results are encouraging because several frequencies and PRIs spreading in the plane characterize them. We note that beyond 40% mixing, the algorithm cannot identify Sets of pulses B and C and confuse them with Emitter D because these emitters have similar characteristics and PRIs.

Same observation as for the experimentation with the outliers presented in Section 5.2.2, using histograms to represent the sets of pulses does not seem suitable. In general, the algorithm confuses the characteristic points of the RADARs with those with average characteristics, i.e., represented in the middle of the plane. Note that despite an identification base with RADARs having very similar characteristics, in Figure 5.21b, the true class comes in 3rd position, and the OT differences are rather small.



(a) Set of pulses A.



(b) Set of pulses C.

Figure 5.21: Identification results for Sets of pulses A and C with a mixing rate of 40% in the $(f_n, dtoa_n)$ plane.

This chapter presents experiments to evaluate the robustness of the deinterleaving methods presented in Sections 3.1 (HACOT) and 3.2 (IHACOT), the identification algorithm presented in Section 4.2 (IDOT) and the modeling of the PRI of Section 4.1.2 by manipulating various parameters (outliers, noise, missing pulses, mixing of pulses). Comparisons were made with other algorithms in the literature; the results showed that the deinterleaving methods provide encouraging results despite the degradation of all these parameters, as does the modeling of the PRI. However, mixed conclusions are made when analyzing the results of the IDOT algorithm during the experiment with pulse mixing and outliers. These observations can be explained by using a database grouping very similar characteristics of emitters to evaluate IDOT. Research directions have been provided to overcome these conclusions, such as using unbalanced optimal transport to make the identification.

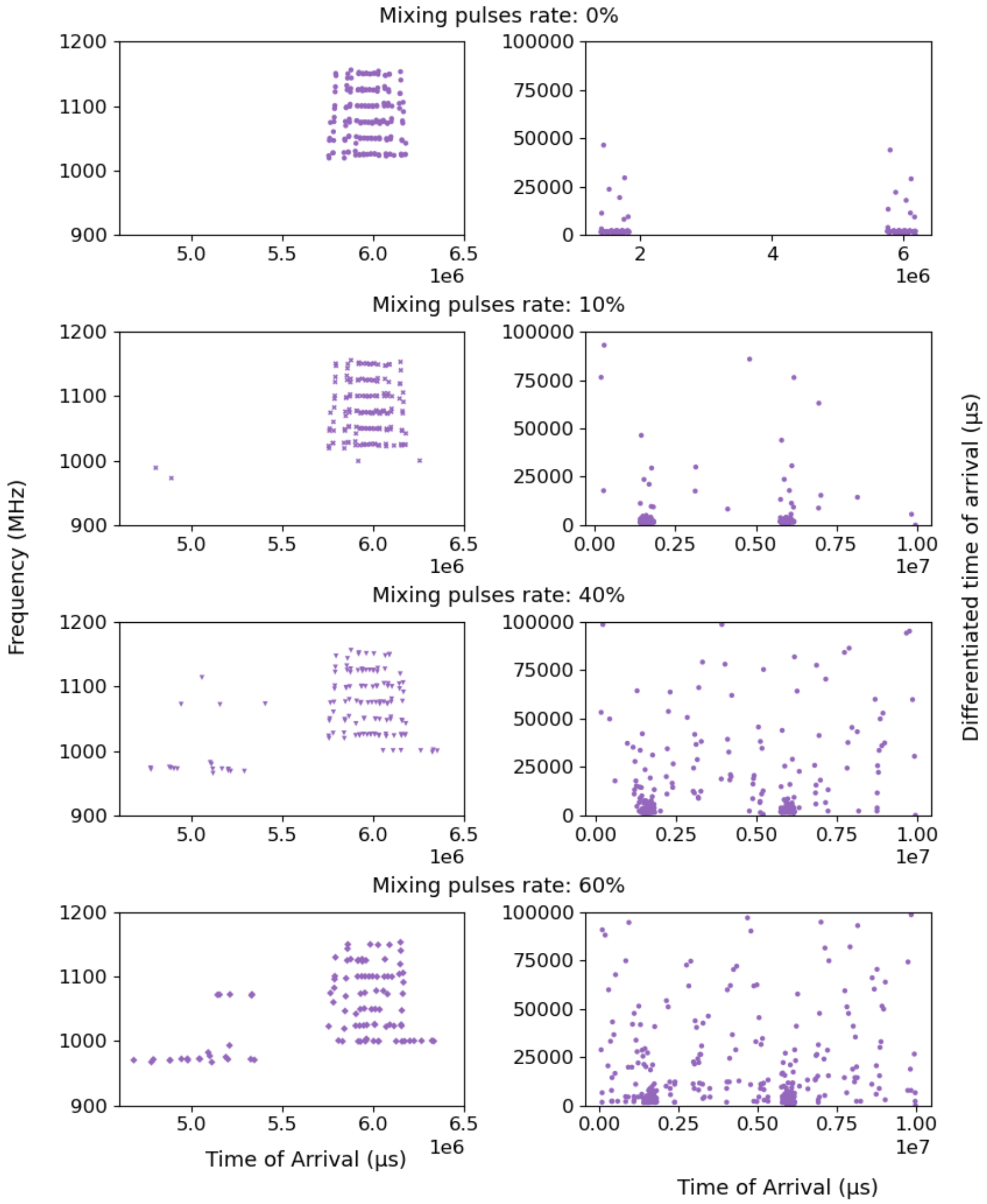


Figure 5.22: Pulse’s representation in the (f_n, pw_n) and $(dtoa_n, toa_n)$ planes according to different mixing rates for Set of pulses A.

Conclusions

General conclusions

This manuscript presents a two-stage RADAR Recognition Process based on the development and use of optimal transport distances with few emitters' features to provide strategic analysis for electronic warfare, followed by results highlighting the robustness of the developed approaches. As a reminder, the RADAR Recognition Process aims to separate and group intercepted pulses from an unknown number of emitters in a signal before identifying them. The opening chapter introduces the necessary tools to understand the developed approaches, followed by a presentation of the RADAR system environment, from its operation throughout the simulation of RADAR data to its modelization. Before going into the methodology, a section reviews the state-of-the-art to summarize the existing methods and underline their limitations and weaknesses.

Two deinterleaving methods are proposed to deal with the first stage of the RADAR recognition process, which can manage complex emitters without considering the PRI pattern. Hierarchical agglomerative clustering combined with optimal transport distances is used to identify emitters using several frequencies, presented agility characteristics, or having multiple operation modes. The assumption is that the pulses of emitters are received at similar times. The HDBSCAN algorithm is applied to frequency and pulse width to separate the pulses into different clusters. As several clusters can represent an emitter, the clusters were grouped using optimal transport distances in a hierarchical agglomerative clustering built from the arrival time weighed by the level. However, using frequency and pulse width in the first clustering step is straightforward, specifically when the listening perimeter concerns a port or an airport, and the signal is noisy, leading to an overlap of the clusters; a second approach, including pre-clustering made from the time of arrival, frequency, and pulse width, is initiated, to separate pulses better, before applying the previous hierarchical agglomerative clustering using optimal transport distances developed to overcome these limitations.

Then, considering that a set of pulses from the previous stage corresponded to a single emitter, the proposed method classifies the emitter with optimal transport in three dimensions, with pulse repetition interval, frequency, and pulse width, in an unsupervised framework. The identification is made by modeling the pulse repetition interval, considering the rate of missing pulses, making the identification algorithm insensitive to pulse losses. The results obtained from the simulated data are very encouraging, and the emitter can be confidently identified while considering many classes. The identification algorithm does not require classifier retraining when adding a new class; it considers many classes to identify, and the results are easily comprehensible. Finally, the method proposes different levels of identification; it can discern the sub-operation modes employed by the emitters.

Empirical simulation results are presented in the last chapter to highlight the robustness concerning complex emission patterns of the RADAR recognition process; first, the deinterleaving methods' robustness is evaluated through several experiments by manipulating the rate of outliers or noise and analyzing results through several metrics. Experiments demonstrated that the methods could handle short and long signals; a single appearance of lobes is sufficient to perform the deinterleaving with conclusive results. Similarly, the identification algorithm was challenged by randomly removing pulses or by increasing the noise level, similar to the part on deinterleaving and analyzing the results returned by the identification algorithm before presenting a proximity analysis between emitters classes, providing a powerful indicator of the confidence level in the results obtained to the ESM operators.

Future directions

This manuscript presents a RADAR recognition process, overcoming the major drawbacks of the methods developed in the literature using the optimal transport theory, providing simple-to-use yet powerful tools. This work has been a foundation to extend the currently developed methodology; several areas for improvement are being studied.

As previously explained, the deinterleaving approaches are based on accessible and reliable features. The deinterleaving results could be improved by including additional features, such as the direction of arrival, which is not always accessible but highly discriminating. A complementary algorithm could be developed based on temporarily available features and combined with the main methodology to improve the deinterleaving of a signal. Finally, applying the hierarchical clustering algorithm using the optimal transport distances assumes that the clusters belonging to an emitter are simultaneously active, which in some instances is not valid.

When setting the confidence threshold to stop the aggregations of the dendrogram during the hierarchical approach, the p-values were sorted to identify the first break to set this threshold. However, in Figures 3.22b, 3.23b and 3.24b, more significant breaks appearing later. It could be interesting to analyze the origin of these breaks.

Deinterleaving methods rely on using the HDBSCAN algorithm to initialize pulse separation. HDBSCAN can identify outliers and thus exclude specific pulses from the analysis. However, pulses from emitters with very low emissions over time may be categorized in this group. A methodological approach could be developed to manage these outliers and check whether they correspond to pulses coming from another emitter.

Additionally, the algorithms are expected to break according to the signal quality, the number of pulses intercepted, and, above all, the similarity of the emitters precisely when the listening perimeter concerns a place where many emitters with identical characteristics are grouped. The similarity between emitters strongly influences the approaches, and it's impossible to distinguish them if they share identical characteristics. Besides, determining the number of emitters in a signal could break the algorithm. In the electronic warfare context, ESM operators deliberately select listening areas; specific areas like airports or ports are not typically favored for these activities, although these locations can potentially pick up many emitters' signals. Our considerations have not accounted for scenarios where a hundred or more emitters might be within the surveillance area. Nevertheless, a potential solution could be to build a semi-supervised approach that involves establishing and employing a reference database within the listening zone to distinguish between recognized emitters, leaving the remaining pulses for further processing.

Concerning the identification, the developed algorithm classifies the known RADAR emitters belonging to a reference database. Detecting the new emitters and automatically enriching this database to propose a complete, unsupervised solution for identifying emitters could be an exciting extension of this work.

As previously detailed, some emitters transmit their PRI randomly. The hypothesis was made that the distribution of their PRI could be represented by a Gaussian distribution and a mixture of Gaussian distributions when the rate of missing pulses is considered. However, in Chapter 4, when the missing pulse rate is high, the algorithm tends to confuse the set of pulse distributions with the distribution of the emitter class having random PRI. When the missing pulse rate is high, the PRI distribution of sets of pulses is characterized by many PRI values, spreading the distribution. In contrast, by construction, the GMM distribution is continuous and characterized by multiple values. The sets of pulse representations with histograms appear to overly smooth out the distributions. Exploring alternative methods to represent the pulse sets could be beneficial, ensuring that the data trends are still effectively captured.

Similar to modeling the PRI in the presence of missing pulses, the frequency distribution could be modeled according to different rates of outliers. This approach would strengthen the results of the identification algorithm because frequency is a very discriminating characteristic.

As for the deinterleaving part, the identification methodology is based on only three reliable and available characteristics; other features describing emitters could be added to complete and strengthen the identification when available.

A basic version of a data simulator was presented to evaluate the results and validate the methods developed. Further consideration could be given to the simulator to enhance its complexity and provide more realistic and complex RADAR signals. The development of real-time algorithms for signal deinterleaving plays a crucial role in electronic warfare, and these algorithms enable the rapid extraction of vital information and electronics to the ESM operators using rapid decision-making capabilities. The different methods developed could be integrated into embedded systems to be tested and evaluated in real-time.

In this research, an assumption was made according to which no adverse measures introducing disruptive signals had been taken into account. However, the interception of signals under certain conditions could return them with spoofing or interference, which can significantly impact signal quality. Further analysis could be done to assess the robustness of these methods against such disruption techniques.

This manuscript proposes new unsupervised methodologies using optimal transport distances to deinterleave a RADAR signal and identify the emitters present. Particular attention was paid to data simulation, a significant element in constructing methods. Several areas for improvement are proposed to continue this research and improve the results. The end of an innovative methodology symbolizes the beginning of the exploration of even more innovative solutions.

Contributions

Articles in peer-reviewed journals

- "Specific Emitter Identification based on Optimal Transport Distances and the Missing Pulses Rate Modelization for the Pulse Repetition of Intervals," in IEEE Transactions on Aerospace and Electronic Systems, 2024. Mottier Manon, Gilles Chardon, and Frédéric Pascal. **(In process).**
- "Deinterleaving RADAR emitters with optimal transport distances," in IEEE Transactions on Aerospace and Electronic Systems, 2024. Mottier Manon, Gilles Chardon, and Frédéric Pascal. Id HAL: hal-04360282.

Conferences with proceedings

- "Désentrelacement et classification des émetteurs RADARs basés sur l'utilisation des distances de Transport Optimal." Conference on Artificial Intelligence for Defense. 2022. Mottier Manon, Gilles Chardon, and Frédéric Pascal. Id HAL: hal-03881736.
- "Désentrelacement et classification de signaux RADAR basés sur des distances de transport optimal." XXVIIIème Colloque Francophone de Traitement du Signal et des Images GRETSI 2022. Mottier Manon, Gilles Chardon, and Frédéric Pascal. Id HAL: hal-03851279.
- "RADAR Emitter Classification with Optimal Transport Distances." 2022 30th European Signal Processing Conference (EUSIPCO). IEEE, 2022. Mottier Manon, Gilles Chardon, and Frédéric Pascal. Id HAL: hal-03869152.
- "Deinterleaving and Clustering unknown RADAR pulses." 2021 IEEE Radar Conference (RadarConf21). IEEE, 2021. Mottier Manon, Gilles Chardon, and Frédéric Pascal. Id HAL: hal-03245067.

Patents

- "Procédé d'identification d'un émetteur radar et système d'identification associé". Mottier Manon, Gilles Chardon, and Frédéric Pascal. 2023/01.
- "Procédé de désentrelacement d'impulsions RADAR". Mottier Manon, Gilles Chardon, and Frédéric Pascal. 2023/01.

Lectures

- "Désentrelacement et classification des émetteurs RADARs basés sur l'utilisation des distances de Transport Optimal." Conference on Artificial Intelligence for Defense (CAID), Rennes (France), 2022.
- "Désentrelacement et classification de signaux RADAR basés sur des distances de transport optimal." XXVIIIème Colloque Francophone de Traitement du Signal et des Images (GRETSI), Grenoble (France), 2022.
- "RADAR Emitter Classification with Optimal Transport Distances." 30th European Signal Processing Conference (EUSIPCO), Belgrade (Serbie), 2022.
- "Radar Emitter Classification with Optimal Transport Distances". 5th edition of SONDRA workshop, Avignon (France), 2022.
- "Deinterleaving and Clustering unknown RADAR pulses." IEEE Radar Conference (RadarConf21), Atlanta (USA), 2021.
- "Deinterleaving and Clustering unknown RADAR pulses". Junior Conference on Data Science and Engineering (JDSE), Gif-sur-Yvette (France), 2021.

Symbols and Notations

Algorithm

DBSCAN	Density-Based Spatial Clustering of Applications with Noise
GMM	Gaussian Mixtures Models
HAC	Hierarchical Agglomerative Clustering
HACOT	Hierarchical Agglomerative Clustering using Optimal Transport distances
HDBSCAN	Hierarchical Density-Based Spatial Clustering of Applications with Noise
HMM	Hidden Markov Model
IDOT	Identification of Emitters using Optimal Transport Distances
IHACOT	Improved Hierarchical Agglomerative Clustering using Optimal Transport distances
KDE	Kernel Density Estimation
MLE	Maximum Likelihood Estimation
OPTICS	Ordering Points to Identify the Clustering Structure

Acronym

ASP	Antenna Scan Pattern
ATP	Antenna Scan Type
ATR	Automatic Target Recognition
AI	Artificial Intelligence
CW	Continuous Wave
ELINT	Electronic Intelligence
EOT	Entropy-regularized Optimal Transport
ESM	Electronic Support Measures
EW	Electronique Warfare

RRP	RADAR Recognition Process
ROEM	Renseignement d'origine électromagnétique
SEI	Specific Emitter Identification
SIGINT	Signals Intelligence
OT	Optimal Transport
PDW)	Pulse Description Word
UOT	Unbalanced Optimal Transport

Feature

doa	Direction of Arrival
f	Frequency
g	Level
pri	Pulse Repetition Interval characteristic
$dtoa$	Pulse Repetition Interval estimated
pw	Pulse Width
snr	Signal to Noise Ratio
toa	Time of Arrival

Mathematical Notation

M_{ari}	Adjusted Rand Index
S_{CH}	Calinski Harabsz Score
M_{cp}	Completeness score
S_{DB}	Davies Bouldin Score
t_{epps}	Epps-Singleton statistic
S_{GAP}	Gap Score
M_{hm}	Homogeneity score
t_{ks}	Kolmogorov-Smirnov statistic
S_{SIL}	Silhouette Score
t_{std}	Student statistic

List of Figures

List of Tables

List of Algorithms

Bibliography

[Ada03] David Adamy. *Introduction to electronic warfare modeling and simulation*. Artech House, 2003.

[Ada14] David Adamy. *Practical Communication Theory*. 2nd. Scitech Pub Inc, 2014. isbn: 1613531869.

[AJ18] David Alvarez-Melis and Tommi S. Jaakkola. "Gromov-Wasserstein Alignment of Word Embedding Spaces". In: *CoRR* abs/1809.00013 (2018). arXiv: `1809.00013`.

[Ank+99] Mihael Ankerst, Markus M Breunig, Hans-Peter Kriegel, and Jörg Sander. "OPTICS: Ordering points to identify the clustering structure". In: *ACM Sigmod record* 28.2 (1999), pp. 49–60.

[AC21] Sabine Apfeld and Alexander Charlish. "Recognition of Unknown Radar Emitters With Machine Learning". In: *IEEE Transactions on Aerospace and Electronic Systems* 57.6 (2021), pp. 4433–4447. doi: `10.1109/TAES.2021.3098125`.

[ACB17] Martin Arjovsky, Soumith Chintala, and Léon Bottou. "Wasserstein generative adversarial networks". In: *International conference on machine learning*. PMLR. 2017, pp. 214–223.

[AA07] AW Ata'a and SN Abdullah. "Deinterleaving of radar signals and PRF identification algorithms". In: *IET radar, sonar & navigation* 1.5 (2007), pp. 340–347.

[BE12] Billur Barshan and Bahaeddin Eravci. "Automatic Radar Antenna Scan Type Recognition in Electronic Warfare". In: *IEEE Transactions on Aerospace and Electronic Systems* 48.4 (2012), pp. 2908–2931. doi: `10.1109/TAES.2012.6324669`.

[Ben03] Jean-David Benamou. "Numerical resolution of an "unbalanced" mass transport problem". In: *ESAIM: Mathematical Modelling and Numerical Analysis* 37.5 (2003), pp. 851–868.

[Bon+11] Nicolas Bonneel, Michiel Van De Panne, Sylvain Paris, and Wolfgang Heidrich. "Displacement interpolation using Lagrangian mass transport". In: *Proceedings of the 2011 SIGGRAPH Asia conference*. 2011, pp. 1–12.

[BA97] Adrian W Bowman and Adelchi Azzalini. *Applied smoothing techniques for data analysis: the kernel approach with S-Plus illustrations*. Vol. 18. OUP Oxford, 1997.

[CH74] Tadeusz Caliński and Jerzy Harabasz. "A dendrite method for cluster analysis". In: *Communications in Statistics-theory and Methods* 3.1 (1974), pp. 1–27.

[Cam+15] Ricardo J. G. B. Campello, Davoud Moulavi, Arthur Zimek, and Jörg Sander. "Hierarchical Density Estimates for Data Clustering, Visualization, and Outlier Detection". In: *ACM Trans. Knowl. Discov. Data* 10.1 (July 2015). issn: 1556-4681. doi: `10.1145/2733381`.

[CMS13] Ricardo JGB Campello, Davoud Moulavi, and Jörg Sander. "Density-based clustering based on hierarchical density estimates". In: *Pacific-Asia conference on knowledge discovery and data mining*. Springer. 2013, pp. 160–172.

[CPD20] Saptarshi Chakraborty, Debolina Paul, and Swagatam Das. "Hierarchical clustering with optimal transport". In: *Statistics & Probability Letters* (2020), p. 108781.

[Cha+06] Yiu Tong Chan, B Haynes Lee, R Inkol, and F Chan. "Estimation of pulse parameters by convolution". In: *2006 Canadian Conference on Electrical and Computer Engineering*. IEEE. 2006, pp. 17–20.

[Cha+10] YT Chan, BH Lee, R Inkol, and F Chan. "Estimation of pulse parameters by autoconvolution and least squares". In: *IEEE transactions on aerospace and electronic systems* 46.1 (2010), pp. 363–374.

[Chi+15] Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. "Unbalanced Optimal Transport: Geometry and Kantorovich Formulation". working paper or preprint. Aug. 2015. url: `https://hal.science/hal-01271981`.

[CR93] Vaughan Clarkson and Pinaki S Ray. *Parallel Processing for Differencing and Histogramming*. Tech. rep. ELECTRONICS RESEARCH LAB ADELAIDE (AUSTRALIA), 1993.

[CM98] T. Conroy and J.B. Moore. "The limits of extended Kalman filtering for pulse train deinterleaving". In: *IEEE Transactions on Signal Processing* 46.12 (1998), pp. 3326–3332. doi: `10.1109/78.735307`.

[Cou+17] Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. "Joint distribution optimal transportation for domain adaptation". In: *Advances in neural information processing systems* 30 (2017).

[Cut13] Marco Cuturi. “Sinkhorn distances: Lightspeed computation of optimal transport”. In: *Advances in neural information processing systems* 26 (2013).

[DH82] CL Davies and P Hollands. “Automatic processing for ESM”. In: *IEE Proceedings F (Communications, Radar and Signal Processing)*. Vol. 129. 3. IET. 1982, pp. 164–171.

[DB79] David L Davies and Donald W Bouldin. “A cluster separation measure”. In: *IEEE transactions on pattern analysis and machine intelligence* 2 (1979), pp. 224–227.

[DLR77] Arthur P Dempster, Nan M Laird, and Donald B Rubin. “Maximum likelihood from incomplete data via the EM algorithm”. In: *Journal of the royal statistical society: series B (methodological)* 39.1 (1977), pp. 1–22.

[DPR18] Arnaud Dessein, Nicolas Papadakis, and Jean-Luc Rouas. “Regularized optimal transport and the rot mover’s distance”. In: *Journal of Machine Learning Research* 19.15 (2018), pp. 1–53.

[Din+18] Lida Ding, Shilian Wang, Fanggang Wang, and Wei Zhang. “Specific Emitter Identification via Convolutional Neural Networks”. In: *IEEE Communications Letters* 22.12 (2018), pp. 2591–2594. doi: `10.1109/LCOMM.2018.2871465`.

[DLW22] Xin Dong, Yan Liang, and Jie Wang. “Distributed Clustering Method Based on Spatial Information”. In: *IEEE Access* 10 (2022), pp. 53143–53152. doi: `10.1109/ACCESS.2022.3175879`.

[DGK18] Pavel Dvurechensky, Alexander Gasnikov, and Alexey Kroshnin. “Computational optimal transport: Complexity by accelerated gradient descent is better than by Sinkhorn’s algorithm”. In: *International conference on machine learning*. PMLR. 2018, pp. 1367–1376.

[ES86] TW Epps and Kenneth J Singleton. “An omnibus test for the two-sample problem using the empirical characteristic function”. In: *Journal of Statistical Computation and Simulation* 26.3-4 (1986), pp. 177–203.

[Est+96] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. “A density-based algorithm for discovering clusters in large spatial databases with noise.” In: *Kdd*. Vol. 96. 34. 1996, pp. 226–231.

[FS08] Alfonso Farina and M Skolnik. “Electronic counter-countermeasures”. In: *Radar handbook*. Vol. 2. McGraw-Hill New York, NY, USA, 2008.

[Fat+21] Kilian Fatras, Thibault Séjourné, Rémi Flamary, and Nicolas Courty. “Unbalanced minibatch optimal transport; applications to domain adaptation”. In: *International Conference on Machine Learning*. PMLR. 2021, pp. 3186–3197.

[Fla+21] Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, Léo Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer. “POT: Python Optimal Transport”. In: *Journal of Machine Learning Research* 22.78 (2021), pp. 1–8. url: `http://jmlr.org/papers/v22/20-451.html`.

[FD81] David Freedman and Persi Diaconis. “On the histogram as a density estimator: L 2 theory”. In: *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete* 57.4 (1981), pp. 453–476.

[FH73] Keinosuke Fukunaga and L Hostetler. “Optimization of k nearest neighbor density estimates”. In: *IEEE Transactions on Information Theory* 19.3 (1973), pp. 320–326.

[GM96] Wilfrid Gangbo and Robert J McCann. “The geometry of optimal transportation”. In: (1996).

[Gas+20] Stefano Gasperini, Magdalini Paschali, Carsten Hopke, David Wittmann, and Nassir Navab. “Signal Clustering With Class-Independent Segmentation”. In: *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2020, pp. 3982–3986. doi: `10.1109/ICASSP40776.2020.9053409`.

[Ge+19] Zhipeng Ge, Xian Sun, Wenjuan Ren, Wenbin Chen, and Guangluan Xu. “Improved Algorithm of Radar Pulse Repetition Interval Deinterleaving Based on Pulse Correlation”. In: *IEEE Access* 7 (2019), pp. 30126–30134. doi: `10.1109/ACCESS.2019.2901013`.

[Gen+21] Zhe Geng, He Yan, Jindong Zhang, and Daiyin Zhu. “Deep-Learning for Radar: A Survey”. In: *IEEE Access* 9 (2021), pp. 141800–141818. doi: `10.1109/ACCESS.2021.3119561`.

[GMW08] Xuhua Gong, Huadong Meng, and Xiqin Wang. “A GMM-based Algorithm for Classification of Radar emitters”. In: *2008 9th International Conference on Signal Processing*. 2008, pp. 2434–2437. doi: `10.1109/ICOSP.2008.4697641`.

[Hal+91] Peter Hall, Simon J Sheather, MC Jones, and James Stephen Marron. “On optimal data-based bandwidth selection in kernel density estimation”. In: *Biometrika* 78.2 (1991), pp. 263–269.

[HW79] John A Hartigan and Manchek A Wong. “Algorithm AS 136: A k-means clustering algorithm”. In: *Journal of the royal statistical society. series c (applied statistics)* 28.1 (1979), pp. 100–108.

[Hoc98] Melinda Hock. “Kalman filter predictor and initialization algorithm for PRI tracking”. In: *Transaction of United States Naval Research Laboratory* (1998).

[Hoi80] David B Hoisington. *Electronic Warfare*. Naval Postgraduate School, 1980.

[HA85] Lawrence Hubert and Phipps Arabie. “Comparing partitions”. In: *Journal of classification* 2 (1985), pp. 193–218.

[JM15] Richard Jensen and Neil Mac Parthaláin. “Towards scalable fuzzy–rough feature selection”. In: *Information Sciences* 323 (2015), pp. 1–15.

[Joh67] Stephen C Johnson. “Hierarchical clustering schemes”. In: *Psychometrika* 32.3 (1967), pp. 241–254.

[Kan+23] Zhi Kang, Yi Zhong, Yaoyun Wu, and Yihong Cai. “Signal Deinterleaving Based on U-Net Networks”. In: *2023 8th International Conference on Computer and Communication Systems (ICCCS)*. 2023, pp. 62–67. doi: `10.1109/ICCCS57501.2023.10150837`.

[KO04] A. Kawalec and R. Owczarek. “Radar emitter recognition using intrapulse data”. In: *15th International Conference on Microwaves, Radar and Wireless Communications (IEEE Cat. No.04EX824)*. Vol. 2. 2004, 435–438 Vol.2. doi: `10.1109/MIKON.2004.1357059`.

[KGG85] James M. Keller, Michael R. Gray, and James A. Givens. “A fuzzy K-nearest neighbor algorithm”. In: *IEEE Transactions on Systems, Man, and Cybernetics* SMC-15.4 (1985), pp. 580–585. doi: `10.1109/TSMC.1985.6313426`.

[Kru13] John K Kruschke. “Bayesian estimation supersedes the t test.” In: *Journal of Experimental Psychology: General* 142.2 (2013), p. 573.

[Les+23] Louis Lesieur, Jean-Marc Le Caillec, Ali Khenchaf, and Vincent Guardia. “Evaluation of a two-step clustering method for radar pulse deinterleaving”. In: *2023 IEEE Conference on Antenna Measurements and Applications (CAMA)*. 2023, pp. 1047–1051. doi: `10.1109/CAMA57522.2023.10352731`.

[LY14] Jingchao Li and Yulong Ying. “Radar signal recognition algorithm based on entropy theory”. In: *The 2014 2nd International Conference on Systems and Informatics (ICSAI 2014)*. 2014, pp. 718–723. doi: `10.1109/ICSAI.2014.7009379`.

[LLH20] Xueqiong Li, Zhangmeng Liu, and Zhitao Huang. “Deinterleaving of Pulse Streams With Denoising Autoencoders”. In: *IEEE Transactions on Aerospace and Electronic Systems* 56.6 (2020), pp. 4767–4778. doi: `10.1109/TAES.2020.3004208`.

[LMS18] Matthias Liero, Alexander Mielke, and Giuseppe Savaré. “Optimal entropy-transport problems and a new Hellinger–Kantorovich distance between positive measures”. In: *Inventiones mathematicae* 211.3 (2018), pp. 969–1117.

[Liu+05] Jun Liu, J.P.Y. Lee, Lingjie Li, Zhi-Quan Luo, and K.M. Wong. “Online clustering algorithms for radar emitter classification”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.8 (2005), pp. 1185–1196. doi: `10.1109/TPAMI.2005.166`.

[Liu21] Zhang-Meng Liu. “Pulse Deinterleaving for Multifunction Radars With Hierarchical Deep Neural Networks”. In: *IEEE Transactions on Aerospace and Electronic Systems* 57.6 (2021), pp. 3585–3599. doi: `10.1109/TAES.2021.3079571`.

[LY19] Zhang-Meng Liu and Philip S. Yu. “Classification, Denoising, and Deinterleaving of Pulse Streams With Recurrent Neural Networks”. In: *IEEE Transactions on Aerospace and Electronic Systems* 55.4 (2019), pp. 1624–1639. doi: `10.1109/TAES.2018.2874139`.

[LK98] A. Logothetis and V. Krishnamurthy. “An interval-amplitude algorithm for deinterleaving stochastic pulse train sources”. In: *IEEE Transactions on Signal Processing* 46.5 (1998), pp. 1344–1350. doi: `10.1109/78.668796`.

[LK07] Jarmo Lunden and Visa Koivunen. “Automatic Radar Waveform Recognition”. In: *IEEE Journal of Selected Topics in Signal Processing* 1.1 (2007), pp. 124–136. doi: 10.1109/JSTSP.2007.897055.

[MR79] YP Mack and Murray Rosenblatt. “Multivariate k-nearest neighbor density estimates”. In: *Journal of Multivariate Analysis* 9.1 (1979), pp. 1–15.

[Mac+67] James MacQueen et al. “Some methods for classification and analysis of multivariate observations”. In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[MA98] Malik Magdon-Ismail and Amir Atiya. “Neural networks for density estimation”. In: *Advances in Neural Information Processing Systems* 11 (1998).

[MP11] Ala Mahdavi and Amir Mansour Pezeshk. “A Fast Enhanced Algorithm of PRI Transform”. In: *2011 Sixth International Symposium on Parallel Computing in Electrical Engineering*. 2011, pp. 179–184. doi: 10.1109/PARELEC.2011.20.

[Mao+09] Yan Mao, Jun Han, Guohua Guo, and Xu Qing. “An Improved Algorithm of PRI Transform”. In: *2009 WRI Global Congress on Intelligent Systems*. Vol. 3. 2009, pp. 145–149. doi: 10.1109/GCIS.2009.313.

[Mar89] HK Mardia. “New techniques for the deinterleaving of repetitive sequences”. In: *IEE Proceedings F (Radar and Signal Processing)*. Vol. 136. 4. IET. 1989, pp. 149–154.

[Mas51] Frank J Massey Jr. “The Kolmogorov-Smirnov test for goodness of fit”. In: *Journal of the American statistical Association* 46.253 (1951), pp. 68–78.

[MB88] Geoffrey J McLachlan and Kaye E Basford. *Mixture models: Inference and applications to clustering*. Vol. 38. M. Dekker New York, 1988.

[MK07] Geoffrey J McLachlan and Thriyambakam Krishnan. *The EM algorithm and extensions*. John Wiley & Sons, 2007.

[Mém11] Facundo Mémoli. “Gromov–Wasserstein distances and the metric approach to object matching”. In: *Foundations of computational mathematics* 11 (2011), pp. 417–487.

[MP92] DJ Milojević and BM Popović. “Improved algorithm for the deinterleaving of radar pulses”. In: *IEE Proceedings F (Radar and Signal Processing)*. Vol. 139. 1. IET. 1992, pp. 98–104.

[Mon81] Gaspard Monge. “Mémoire sur la théorie des déblais et des remblais”. In: *Mem. Math. Phys. Acad. Royale Sci.* (1781), pp. 666–704.

[MK94] J.B. Moore and V. Krishnamurthy. “Deinterleaving pulse trains using discrete-time stochastic dynamic-linear models”. In: *IEEE Transactions on Signal Processing* 42.11 (1994), pp. 3092–3103. doi: `10.1109/78.330369`.

[Nel93] D. Nelson. “Special purpose correlation functions for improved signal detection and parameter estimation”. In: *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 4. 1993, 73–76 vol.4. doi: `10.1109/ICASSP.1993.319597`.

[Ner06] Filippo Neri. *Introduction to electronic defense systems*. SciTech Publishing, 2006.

[Nis83] K Nishiguchi. “A new method for estimation of pulse repetition intervals”. In: *National convention record of iece of japan*. 1983.

[NK00] K. Nishiguchi and M. Kobayashi. “Improved algorithm for estimating pulse repetition intervals”. In: *IEEE Transactions on Aerospace and Electronic Systems* 36.2 (2000), pp. 407–421. doi: `10.1109/7.845217`.

[Noo99] G.P. Noone. “A neural approach to automatic pulse repetition interval modulation recognition”. In: *1999 Information, Decision and Control. Data and Information Fusion Symposium, Signal Processing and Communications Symposium and Decision and Control Symposium. Proceedings (Cat. No.99EX251)*. 1999, pp. 213–218. doi: `10.1109/IDC.1999.754156`.

[NAA20] Mustafa Atahan Nuhoglu, Yasar Kemal Alp, and Fatih Cagatay Akyon. “Deep Learning for Radar Signal Detection in Electronic Warfare Systems”. In: *2020 IEEE Radar Conference (RadarConf20)*. 2020, pp. 1–6. doi: `10.1109/RadarConf2043947.2020.9266381`.

[Nuh+23] Mustafa Atahan Nuhoglu, Yasar Kemal Alp, Mehmet Ege Can Ulusoy, and Hakan Ali Cirpan. “Image Segmentation for Radar Signal Deinterleaving Using Deep Learning”. In: *IEEE Transactions on Aerospace and Electronic Systems* 59.1 (2023), pp. 541–554. doi: `10.1109/TAES.2022.3188225`.

[Par62] Emanuel Parzen. “On estimation of a probability density function and mode”. In: *The annals of mathematical statistics* 33.3 (1962), pp. 1065–1076.

[PCS16] Gabriel Peyré, Marco Cuturi, and Justin Solomon. “Gromov-wasserstein averaging of kernel and distance matrices”. In: *International conference on machine learning*. PMLR. 2016, pp. 2664–2672.

[Poi12] Richard Poisel. *Electronic warfare target location methods*. Artech House, 2012.

[RP15] Julien Rabin and Nicolas Papadakis. “Convex color image segmentation with optimal transport distances”. In: *Scale Space and Variational Methods in Computer Vision: 5th International Conference, SSVM 2015, Lège-Cap Ferret, France, May 31-June 4, 2015, Proceedings 5*. Springer. 2015, pp. 256–269.

[Rac85] Svetlozar T Rachev. “The Monge–Kantorovich mass transference problem and its stochastic applications”. In: *Theory of Probability & Its Applications* 29.4 (1985), pp. 647–676.

[Ray98] P.S. Ray. “A novel pulse TOA analysis technique for radar identification”. In: *IEEE Transactions on Aerospace and Electronic Systems* 34.3 (1998), pp. 716–721. doi: `10.1109/7.705881`.

[RME18] Guillaume Revillon, Ali Mohammad-Djafari, and Cyrille Enderli. “Radar emitters classification and clustering with a scale mixture of normal distributions”. In: *2018 IEEE Radar Conference (RadarConf18)*. 2018, pp. 1371–1376. doi: `10.1109/RADAR.2018.8378764`.

[RR10] Brian D. Rigling and Craig Roush. “ACF-based classification of phase modulated waveforms”. In: *2010 IEEE Radar Conference*. 2010, pp. 287–291. doi: `10.1109/RADAR.2010.5494610`.

[RH07] Andrew Rosenberg and Julia Hirschberg. “V-measure: A conditional entropy-based external cluster evaluation measure”. In: *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*. 2007, pp. 410–420.

[Ros56] Murray Rosenblatt. “Remarks on some nonparametric estimates of a density function”. In: *The annals of mathematical statistics* (1956), pp. 832–837.

[Rou87] Peter J Rousseeuw. “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis”. In: *Journal of computational and applied mathematics* 20 (1987), pp. 53–65.

[RTG00] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. "The earth mover's distance as a metric for image retrieval". In: *International journal of computer vision* 40 (2000), pp. 99–121.

[SR21] Matthew Scherreik and Brian Rigling. "Online Estimation of Radar Emitter Cardinality via Bayesian Nonparametric Clustering". In: *IEEE Transactions on Aerospace and Electronic Systems* 57.6 (2021), pp. 3791–3800. doi: `10.1109/TAES.2021.3103582`.

[Sch80] D Curtis Schleher. *Automatic detection and radar data processing*. Artech House Dedham, MA, 1980.

[Sch99] D Curtis Schleher. *Electronic warfare in the information age*. Artech House, Inc., 1999.

[Sch86] D Curtis Schleher. "Introduction to electronic warfare". In: *Dedham* (1986).

[Sch74] Ralph O. Schmidt. "On Separating Interleaved Pulse Trains". In: *IEEE Transactions on Aerospace and Electronic Systems* AES-10.1 (1974), pp. 162–166. doi: `10.1109/TAES.1974.307812`.

[SS15] Bernhard Schmitzer and Christoph Schnörr. "Globally optimal joint image segmentation and shape matching based on Wasserstein modes". In: *Journal of Mathematical Imaging and Vision* 52 (2015), pp. 436–458.

[SB23] Stefan Scholl and Stefan Brüggenwirth. "Incremental Deinterleaving of Radar Emitters". In: *IEEE Aerospace and Electronic Systems Magazine* 38.8 (2023), pp. 26–36. doi: `10.1109/MAES.2023.3268020`.

[Sco15] David W Scott. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015.

[Sco79] David W Scott. "On optimal and data-based histograms". In: *Biometrika* 66.3 (1979), pp. 605–610.

[STT80] David W Scott, Richard A Tapia, and James R Thompson. "Nonparametric probability density estimation by discrete maximum penalized-likelihood criteria". In: *The annals of statistics* (1980), pp. 820–832.

[SS22] Purabi Sharma and Kandarpa Kumar Sarma. "PRI Modulation Classification in EW Systems Using Deep Learning". In: *International Conference on Emerging Global Trends in Engineering and Technology*. Springer. 2022, pp. 453–462.

[SJ91] Simon J Sheather and Michael C Jones. “A reliable data-based bandwidth selection method for kernel density estimation”. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 53.3 (1991), pp. 683–690.

[Sil86] Bernard W Silverman. *Density estimation for statistics and data analysis*. Vol. 26. CRC press, 1986.

[Sko08] Merrill I Skolnik. *Radar handbook*. McGraw-Hill Education, 2008.

[Sol+15] Justin Solomon, Fernando De Goes, Gabriel Peyré, Marco Cuturi, Adrian Butscher, Andy Nguyen, Tao Du, and Leonidas Guibas. “Convolutional wasserstein distances: Efficient optimal transportation on geometric domains”. In: *ACM Transactions on Graphics (ToG)* 34.4 (2015), pp. 1–11.

[Stu08] Student. “The probable error of a mean”. In: *Biometrika* 6.1 (1908), pp. 1–25.

[Vil09] Cédric Villani. *Optimal transport: old and new*. Vol. 338. Springer, 2009.

[WWZ17] Chao Wang, Jian Wang, and Xudong Zhang. “Automatic radar waveform recognition based on time-frequency analysis and convolutional neural network”. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2017, pp. 2437–2441. doi: `10.1109/ICASSP.2017.7952594`.

[Wan+20] Shiqiang Wang, Caiyun Gao, Qin Zhang, Veerendra Dakulagi, Huiyong Zeng, Guimei Zheng, Juan Bai, Yuwei Song, Jiliang Cai, and Binfeng Zong. “Research and Experiment of Radar Signal Support Vector Clustering Sorting Based on Feature Extraction and Feature Selection”. In: *IEEE Access* 8 (2020), pp. 93322–93334. doi: `10.1109/ACCESS.2020.2993270`.

[WB19] Jonathan Weed and Quentin Berthet. “Estimation of smooth densities in Wasserstein distance”. In: *conference on Learning Theory*. PMLR. 2019, pp. 3118–3119.

[Wil06] Richard Wiley. *ELINT: The interception and analysis of radar signals*. Artech, 2006.

[WW85] DR Wilkinson and AW Watson. “Use of metric techniques in ESM data processing”. In: *IEE Proceedings F (Communications, Radar and Signal Processing)*. Vol. 132. 4. IET. 1985, pp. 229–232.

[Xi+17] Yin Xi, Yingchun Wu, Xiongjun Wu, and Kaichuang Jiang. "An improved SDIF algorithm for anti-radiation radar using dynamic sequence search". In: *2017 36th Chinese Control Conference (CCC)*. 2017, pp. 5596–5601. doi: `10.23919/ChiCC.2017.8028245`.

[XSZ23] Haoran Xiang, Furao Shen, and Jian Zhao. "Deep ToA Mask-Based Recursive Radar Pulse Deinterleaving". In: *IEEE Transactions on Aerospace and Electronic Systems* 59.2 (2023), pp. 989–1006. doi: `10.1109/TAES.2022.3193948`.

[Yan+23] Shuyuan Yang, Xinyi Zhao, Huiling Liu, Chen Yang, Tongqing Peng, Rundong Li, and Feng Zhang. "Deep Contrastive Clustering for Signal Deinterleaving". In: *IEEE Transactions on Aerospace and Electronic Systems* (2023), pp. 1–11. doi: `10.1109/TAES.2023.3322971`.

[Zen+11] D Zeng, X Zeng, G Lu, and B Tang. "Automatic modulation classification of radar signals using the generalised time-frequency representation of Zhao, Atlas and Marks". In: *IET radar, sonar & navigation* 5.4 (2011), pp. 507–516.

[ZLH17] Xiaonan Zhang, Pengcheng Luo, and Xinwu Hu. "A hybrid method for classification and identification of emitter signals". In: *2017 4th International Conference on Systems and Informatics (ICSAI)*. 2017, pp. 1060–1065. doi: `10.1109/ICSAI.2017.8248442`.

[Zho+18] Zhiwen Zhou, Gaoming Huang, Haiyang Chen, and Jun Gao. "Automatic radar waveform recognition based on deep convolutional denoising auto-encoders". In: *Circuits, Systems, and Signal Processing* 37.9 (2018), pp. 4034–4048.

[ZWL22] Mengtao Zhu, Shafei Wang, and Yunjie Li. "Model-Based Representation and Deinterleaving of Mixed Radar Pulse Sequences With Neural Machine Translation Network". In: *IEEE Transactions on Aerospace and Electronic Systems* 58.3 (2022), pp. 1733–1752. doi: `10.1109/TAES.2021.3122411`.