

日本船舶海洋工学会 2022 年度秋季講演会

無人運航船プロジェクト

MEGURI
2040

日本
財団
THE NIPPON
FOUNDATION

DFFAS
Designing the Future of Full Autonomous Ship

完全自律航行船の設計・開発・実証のための DFFAS コンソーシアムのアプローチ — MBSE と MBD を活用した自律航行システムの構築 —

2022/11/18

¹ 株式会社 MTI ² 東京大学大学院 ³ 日本無線株式会社
⁴ 古野電気株式会社 ⁵ 株式会社日本海洋科学 ⁶ 日本郵船株式会社

暮田留依¹, 中島拓也², 樋口穰司³, 西山尚材³, 柳原智哉⁴, 櫻井美奈⁵, 西村遥⁶, 沓名弘二⁶, 中村純¹

目次

1. はじめに： DFFAS プロジェクトの概要
2. 完全自律航行システム開発のアプローチ：モデルを活用した開発フロー
3. 設計フロー：システムの要求洗い出しとモデル構築
4. 検証フロー：モデルを起点としたシミュレーションの活用
5. おわりに



1. はじめに： DFFAS プロジェクトの概要
2. 完全自律航行システム開発のアプローチ：モデルを活用した開発フロー
3. 設計フロー：システムの要求洗い出しとモデル構築
4. 検証フロー：モデルを起点としたシミュレーションの活用
5. おわりに

日本財団 MEGURI2040 プロジェクトと DFFAS コンソーシアム

- 背景：内航海運を取り巻く環境 → ヒューマンファクターを補完する運航の高度化・自律化が必要
 - ▶ 船員の高齢化
 - ▶ 海難事故：約 8 割がヒューマンエラーに起因
- 日本財団 無人運航船プロジェクト MEGURI2040
 - ▶ 2022 年までに無人運航船の実証実験を実施 → 無人運航船の実用化による内航海運の課題解決を目指す
- DFFAS (Designing the Future of Full Autonomous Ship) コンソーシアム
 - ▶ NYK グループを含む国内約 30 社 (国外協力企業を含むと約 60 社) による Open Innovation 体制
 - ▶ MEGURI2040 「無人運航船の実証実験にかかる技術開発共同プログラム」に参画



DFFAS コンソーシアム 参画企業 (国内)

DFFAS コンソーシアムの取り組みと本講演の対象

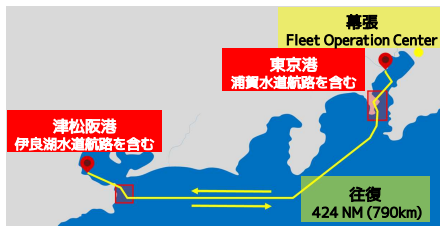
- DFFAS コンソーシアムの成果：幅狭海域を含む完全自律航行に対応したシステムの構築と実証
- APExS-auto： **A**ction **P**laning and **E**xecution **S**ystem for full **auto**nomous
 - ▶ DFFAS コンソーシアムで開発した完全自律航行システムの概念設計
 - ▶ ClassNK および BV より基本設計承認（AiP）取得 @ 2022/03

無人運航船プロジェクト
MEGURI
2040

日本財団
THE NIPPON
FOUNDATION

DFFAS
Designing the Future of Full Autonomous Ship

実証航行のイメージ



完全自律航行システム
APExS-auto



DFFAS における実証航行のイメージと陸側・船側システムの概要

- 本講演の対象： DFFAS における完全自律航行システム（APExS-auto）設計・検証のアプローチ



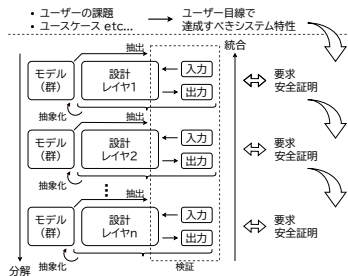
1. はじめに： DFFAS プロジェクトの概要
2. 完全自律航行システム開発のアプローチ：モデルを活用した開発フロー
3. 設計フロー：システムの要求洗い出しとモデル構築
4. 検証フロー：モデルを起点としたシミュレーションの活用
5. おわりに

大規模システムとしての自律航行システムとモデルの必要性

- APEX-auto の特徴：深い機能階層，複雑な相互作用（機械-機械，機械-人） etc.
→ System of Systems (SoS) として構成される自律システム
- SoS の開発範囲特定や設計・検証過程で適切にシステム分割・統合を行うのに必要なのは...
 - ▶ 設計・検証の階層間でのシステム仕様や特性等の網羅性・連続性担保
 - ▶ 多数の関係者間での共通言語によるシステム仕様や特性の表現：モデル の導入

APEX-auto サブシステム構成

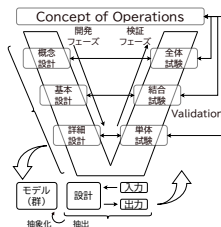
サブシステム名	役割
Maneuvering	航行計画を策定し 船の動きを制御
Propulsion	動力・電力を管理
Communication	船陸間通信
Central Information Management (CIM)	システムの状態を管理



階層的な SoS 構築フローとモデルの活用

モデルを活用したシステム/SoS 開発と V-Process の考え方

- モデル：システムの仕様や特性の共通言語（SysML 等）による抽象表現
 - ▶ SoS のあり様を示す体系的な仕様書
 - ▶ 開発での活用：MBSE (**M**odel **B**ased **S**ystems **E**ngineering), MBD (**M**odel **B**ased **D**esign)
- V-Process：階層単位でシステムの設計と検証を対で捉える開発工程のモデル
 - ▶ ConOps (**C**oncept of **O**perations)：システム享受者の目線で記述される上位要求
 - ▶ 設計：要求に従い，システムのスコップや階層を逐次分割しながらコンポーネントを構築
 - ▶ 検証（Verification）：システムが要求に対して正しく実現されているか
 - ▶ 妥当性確認（Validation）：正しいシステムであるかどうか（ConOps との照合）
- V-Process におけるモデルの有効利用
 - ▶ システム機能設計を階層的に透明化
 - ▶ IMO ルールや船級規則等との整合性の根拠：
Goal Based Standards との相性



V-Process におけるモデルの位置付け（イメージ）



1. はじめに： DFFAS プロジェクトの概要
2. 完全自律航行システム開発のアプローチ：モデルを活用した開発フロー
3. 設計フロー：システムの要求洗い出しとモデル構築
4. 検証フロー：モデルを起点としたシミュレーションの活用
5. おわりに

自律航行システムの要求分析：ConOps を起点とした上位設計モデルの導出

■ DFFAS で策定した自律航行システムの ConOps （抜粋）

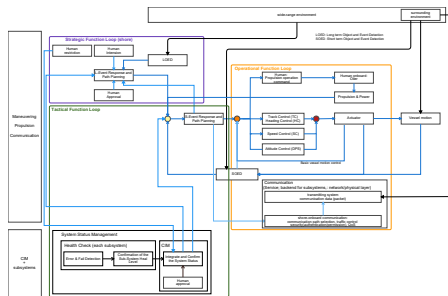
- ▶ 背景と目的：認知・分析・計画・実行の各タスクの自律化と高度支援による内航船の安全性向上
- ▶ 前提
 - ▶ 装備の対象：内航コンテナ船（749GT）
 - ▶ 本船乗組員が介入することなく Berth to Berth の航行が可能（Lloyd 自律船レベル AL4 相当）
- ▶ 機能分担
 - ▶ 人間の役割：陸上での長期計画策定・承認，ステータス承認，遠隔支援
 - ▶ 機械の役割：離棧～着棧までの航行のための認知・分析・計画・実行機能，陸上での高度支援（等）

■ 自律航行システムの上位設計を構築するための要求を ConOps から抽出

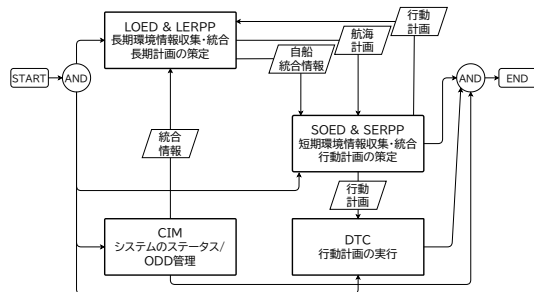
- ▶ 機能の視点
 - ▶ 長期・広範囲，短期・周囲の環境情報を認知しながら航行計画を策定・実行可能
 - ▶ 安全維持：衝突・座礁に至るリスクの抑制
- ▶ 構成の視点
 - ▶ 本船上では機械により一連のタスクを実行
 - ▶ 陸上に要員を配置し支援・承認を実施

自律航行システムの最上位モデル：機能

- MBSE の考え方：機能 = 要求をもとにシステムのふるまいを抽出したモデル
- 機能ブロック：環境とシステムとの相互作用を時間軸に応じて分類
 - ▶ Strategic function：及ぼす影響が長期に渡る機能
 - ▶ Tactical function：Strategic function の出力を遂行するための短期的な機能
 - ▶ Operational function：Tactical function を遂行するための極めて頻繁に繰り返される機能
- 機能フロー：自律航行システム上での計画策定→離棧→着棧 までの一連の流れを表現



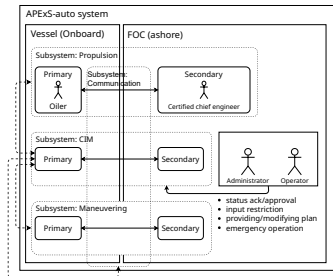
自律航行システムの機能ブロック



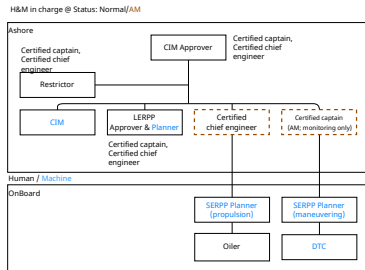
自律航行システムの機能フロー

自律航行システムの最上位モデル：構成

- MBSE の考え方：構成 = 機能の割当先を表現するモデル
 - ConOps からブレークダウンされた要求をもとに，船/陸 と 機械/人 の配置や役割を表現
 - ▶ Strategic function （長期の航海計画策定等）：陸上で人間（陸上船長）の承認のもと機能
 - ▶ Tactical function：船/陸のコンポーネント間で情報同期しながら透過的に機能
 - ▶ Operational function：船上で機能
- サブシステム構成を境界として機能をコンポーネントにマッピング



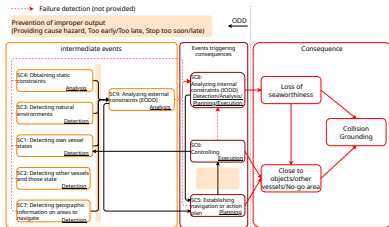
自律航行システムの構成



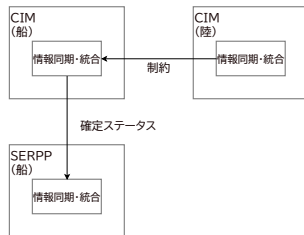
自律航行システムの権限/役割の割当（例）

自律航行システムの最上位モデル：STAMP/STPA による安全性評価の視点

- SoS の安全性：アイテム単位の積み上げでは網羅的・包括的な脅威の影響度評価が困難
- STAMP/STPA：機能間の相互作用を起点とする大規模システムの安全解析手法
 - ▶ 分析対象の系と環境における相互作用をトップダウン的に解析したモデル：コントロールストラクチャ
 - ▶ モデルの粒度：SoS が安全を維持するための制約（安全制約）を実現する視点で設定
- APEX-auto における安全制約の設定
 - ▶ 衝突・座礁：自律航行システムとして回避すべき最終事象
 - ▶ 最終事象回避のために維持されるべき制約群とシーケンス：自律運航船と有人船とのリスク比較の横串



安全制約群の関係

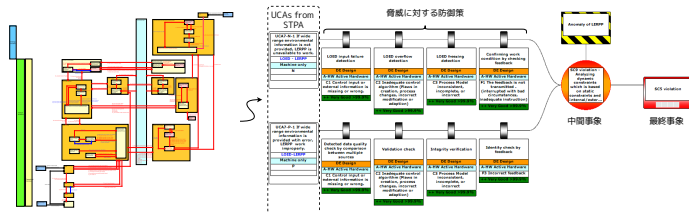


APEX-auto のコントロールストラクチャ（抜粋）

STAMP/STPA と BowTie 分析によるリスクの可視化

■ APEX-auto のアプローチ：STAMP/STPA 分析の統合によるリスクアセスメント

- ▶ BowTie 分析：システムの脅威と中間/最終事象の流れと それらに対する防御策を分析するフレーム
- ▶ コントロールストラクチャをもとに抽出した脅威を起点に BowTie 分析を実施
- ▶ 上位設計モデルを起点としたリスクアセスメント → システムの基本要求抽出 のフローを構築



APEX-auto におけるコントロールストラクチャと BowTie 分析の統合（抜粋）

■ 一連のリスクアセスメント結果：設計や実証オペレーションに反映

- ▶ 防御策：システムに対する安全維持観点での基本要求
- ▶ FMEA：安全制約を軸に影響度の評価 & 対策の策定

V-Process に沿ったモデルベース設計：詳細設計に至る機能要求の洗い出し

■ 上位設計から詳細設計へのブレイクダウン

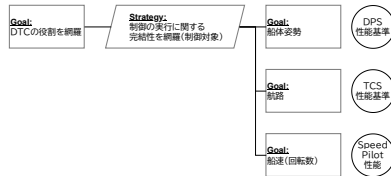
- ▶ 詳細設計：ドメインや機能階層に応じて上位のモデルを連続的に分割・詳細化
- ▶ システム要求の体系的な洗い出しが必要

■ 機能要求抽出と安全論証のフレーム：GSN (Goal Structure Notation)

- ▶ 論証すべき（サブ）ゴールとその論拠（議論の軸）を順次ツリー状に表現：要求の構造化
- ▶ DFFAS における利用：設計として課される要求と機能全体の洗い出しに活用
 - ▶ 機能を網羅するための上位の軸：サブシステム/安全制約（≡ 機能分類）/機能の時間軸



GSN の例：設計要求の洗い出し



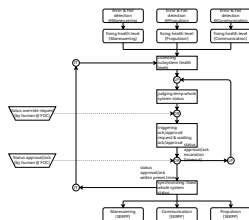
GSN の例：制御機能の洗い出し

モデルの例：システムの状態管理（CIM）

■ CIM（Central Information Management）

- ▶ ODD（**O**perational **D**esign **D**omain）：運行設計領域，自動化/自律化システムが適切に機能する範囲
- ▶ コンポーネント群の健全性情報をもとに自律航行システム全体のステータス＝ODD境界を判断
 - ▶ Normal：正常動作
 - ▶ Active Monitoring：人間の監視を前提にシステムが動作
 - ▶ Remote Fallback：人間がシステムに介入し航行を維持（承認の下でシステム→人に操船権を移譲）
 - ▶ Independent Fallback：システムが安全を維持するための動作

■ 機能フロー図と N2 図により CIM が実行するステータス判定・移行のロジックをモデリング



CIM の機能フロー図

ODD (2 main categories)	Error & fault information warning conditions							
	Warning information warning level (2 main categories)	subsystem health status						
		subsystem health status	subsystem health status					
		subsystem health status	subsystem health status	subsystem health status				
		subsystem health status	subsystem health status	subsystem health status	subsystem health status			
		subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status		
		subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status	
		subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status
		subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status
		subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status	subsystem health status

CIM の N2 図：機能間 I/F の表現



1. はじめに： DFFAS プロジェクトの概要
2. 完全自律航行システム開発のアプローチ：モデルを活用した開発フロー
3. 設計フロー：システムの要求洗い出しとモデル構築
4. 検証フロー：モデルを起点としたシミュレーションの活用
5. おわりに

モデルベースのシステム検証と V-Process

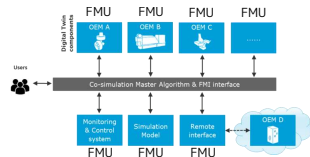
■ V-Process に沿ったモデルベースのシステム検証

- ▶ 下位の検証を網羅的に実施→上位工程での手戻りを抑制可能
- ▶ シミュレーションの活用（バーチャルエンジニアリング）
 - ▶ 検証対象への作用や実環境の入出力を再現可能
- ▶ モデルによるシステムロジックの記述
 - ▶ 機能粒度と検証範囲・境界の対応付け
 - ▶ 柔軟な HIL/MIL テスト
 - ▶ SoS のコミッショニングの効率化

■ FMU (Function Mockup Unit)

- ▶ Open Simulation Platform 上で組み換え可能なモジュールの標準形式
- ▶ 自動車業界のモデルベース開発における活用実績

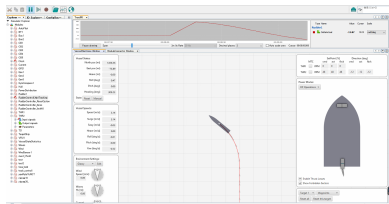
- DFFAS コンソーシアムでの検証のアプローチ：評価対象の機能や周辺環境のモデルを FMU 化
→ FMU や実機を柔軟に統合しながら V-Process を遂行（MIL テスト/HIL テスト）



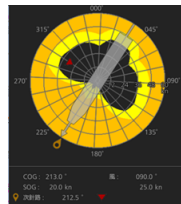
FMU の組合せによる
シミュレーション（イメージ）

シミュレーション環境の活用： MIL テストの事例

- External ODD (EODD)：外部環境要因に基づく自律航行システムの ODD
 - ▶ 自船周囲の環境情報をもとに判定
 - ▶ Maneuvering 健全性情報として CIM に配信→ ODD 内外（ステータス）を判断
 - シミュレーションプラットフォーム上での MIL テストによる EODD 設定の算出
 - ▶ 本船に作用し得る外乱条件（風/波/潮流）のモデル：本船周囲環境の模擬
 - ▶ 航路制御機能のモデル：簡易的な航路追従制御の模擬
 - ▶ 自船運動のモデル：低速域/離着岸操船を含む船体運動の模擬
- 複数の外乱条件の組合せで擬似的な航行を繰り返し実行し判定用の設定値を算出



シミュレーション画面の例

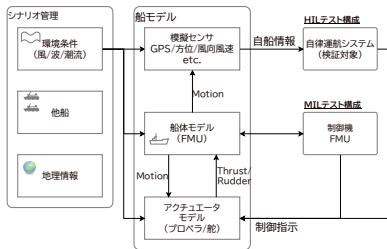


EODD 表示例（古野電気提供）

シミュレーション環境の活用：全体構成と HIL テストの事例

■ 陸上で構築したシミュレーション環境

- ▶ 周辺環境のモデル（例）：他船の AIS 信号を模擬するシナリオ→自動避航機能も評価



シミュレーション環境の構成



シミュレーション検証の様子

■ システムのモデルに基づいた検証項目の抽出 → 30 項目以上の不具合を本船搭載前に解消

- ▶ 自律航行システムの機能シーケンス：正常系シナリオの網羅
- ▶ STPA 分析に基づくコントロールストラクチャ：異常系シナリオの網羅



1. はじめに： DFFAS プロジェクトの概要
2. 完全自律航行システム開発のアプローチ：モデルを活用した開発フロー
3. 設計フロー：システムの要求洗い出しとモデル構築
4. 検証フロー：モデルを起点としたシミュレーションの活用
5. おわりに

まとめ

- 日本財団 無人運航船プロジェクト MEGURI2040 における DFFAS コンソーシアムの取り組み
 - ▶ 完全自律航行システム APEXs-auto の構築と実船による幅狭海域での無人運航実証に成功
- APEXs-auto の構築におけるモデルの活用と V-Process の導入
 - 連続的なシステム要求の洗い出し & マッピング, フロントローディング
 - ▶ ConOps：使用者目線でのシステム特性・スコープ等のドキュメンテーション
 - ▶ 設計フェーズ：ConOps を起点にシステムの要求を洗い出し, 様々な視点からモデルを構築
 - ▶ 検証フェーズ：モデルを起点とした検証項目の抽出とシミュレーション環境構築
- リスクアセスメント：モデルを起点とした大規模システムに対応可能なアプローチの導入
 - ▶ STAMP/STPA と BowTie 分析の統合 → 安全のための要求を抽出し設計にフィードバック



← DFFAS コンソーシアム ドキュメンタリー映像

無人運航船プロジェクト
**MEGURI
2040**

日本財団
THE NIPPON
FOUNDATION

DFFAS
Designing the Future of Maritime Autonomous Ship