

# Bee-gent を用いたバッチプロセスコントロールの構造

(名工大・工) (正)濱口孝司\*・(学)坂本 大・(正)橋本芳宏・(正)伊藤利昭・(正)米谷昭彦・(正)戸苅吉孝

## 1. はじめに

製造と開発を同時にこなせる生産システムとして、自律分散型のバッチ制御システムを提案している[1]。今回は、(株)東芝のマルチエージェントフレームワーク Bee-gent[2, 3, 4]を利用して開発した、マルチバッチやオペレータによる介入も可能な自律分散型運転管理システムを紹介する。

## 2. Bee-gent を用いたシステム構造

Bee-gent では移動エージェントである bee がネットワーク間を移動して非移動エージェントである AW (エージェントラッパー) と KQML ベースの ACL (Agent Communication Language) を用いた XML/ACL を用いて交信する。エージェントラッパーは Java など開発されたアプリケーションプログラムと接続される。図 1 にシステム構成を示した。レシピである bee は Generator で生成され、原料タンクに移動する。その後、レシピに記載された内容に従い、各 AW を通じて設備に機能の実現要求を行う。原料を移送する場合は移送エージェントに移送要求を行う。

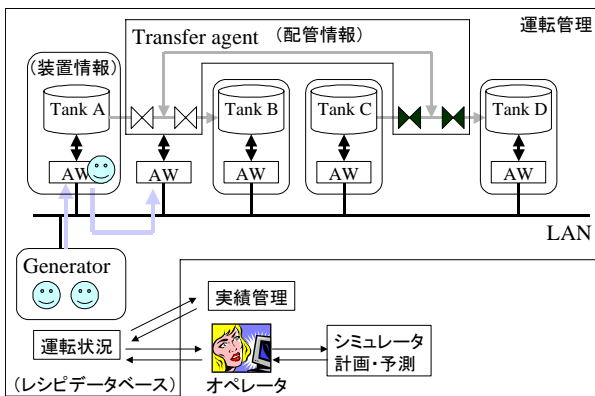


図 1 . システム構成

レシピは図 2 のような状態遷移モデルを持ち、設備などに出す要求は XML にて定義されている。

## 3. マルチスレッドによるマルチバッチや介入の実現

配管設備群を集中管理する移送エージェントは、各レシピの要請に応じて、移送経路と各設備の排他利用ルールを用いてマルチバッチの移送制御を行う。また、各設備や移送コントローラが何らかの処理を行っている時点でオペレータにより情報収集や介入などの操作が必要となる。

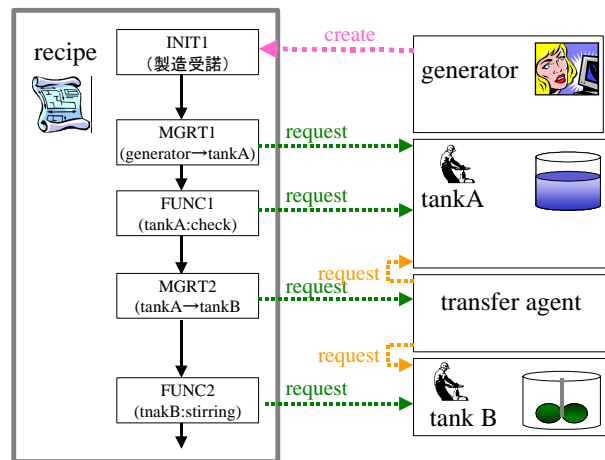


図 2 . レシピの状態遷移モデル

Bee-gent システムでは、1 つの AW を 1 つの bee が排他利用するフレームワークであり、上記の機能を実現できるように、AW が受付をすると bee の要求は一旦完了し、アプリケーションスレッドを発生する方法を用いた。その後、AW は受付待機状態に復帰する。

また、bee は Generator と交信が切断されると消滅してしまうが、スレッド化により bee が消滅しても制御が進行できるようにもなる。しかし、ネットワークが復旧すれば、適切な bee を再投入する必要が生じるため、Generator にオリジナルの bee を残しておき、各設備エージェントに送り込む bee はクローンを用いた。複数原料からなる製品のレシピは複数のクローンを各原料タンクに投入し、原料が合流する時点で後から到着した同一製品用のレシピを消滅させる。

## 参考文献

- 1) Hamaguchi et al , "Autonomous Decentralized Control System for Batch Process" PSE Asia 2002 (2002)
- 2) Bee-gent : Multi-agent framework, <http://www2.toshiba.co.jp/beegent/>
- 3) 川村ら, "Bee-gent : 移動型仲介エージェントによる既存システムの柔軟な活用を目的としたマルチエージェントフレームワーク", 信学論 (D- ), **82**(9), pp.1165-1180(1999)
- 4) 川村ら, "マルチエージェントフレームワーク Bee-gent を用いた電力系統作業停止計画向け分散スケジューリングシステムの開発", 電学論 C, **120**(12), pp.2003-2011(2000)

\* TEL: 052-735-5561, FAX: 052-735-5401  
e-mail: hamaguchi@system.nitech.ac.jp