

リモートペアプログラミング環境における コミュニケーション問題への一対処法

遠藤修一郎[†]

太田剛[‡]

静岡大学大学院情報学研究科[†]

静岡大学情報学部[‡]

1. 背景と目的

2人でプログラミングを行う手法としてペアプログラミングがある。2人のプログラマが1つのマシンでプログラミングを行い、プログラムを実装している人をドライバ、それを見ている人をナビゲータと呼ぶ。ドライバはプログラミングを行い、ナビゲータはデバッグやテストなど相補的な作業を行う。2人のプログラマは常に一緒に作業する。

このようにペアプログラミングを行うためには、2人が同じ時間に同じ場所にいなければならないため、2人が離れている場合には利用することができない。この問題を解決する方法として、マシンを2台用意して2人が離れている状況でもペアプログラミングを行えるリモートペアプログラミングが研究されている[1][2]。また、リモートペアプログラミングを行うためのツールも開発されている[3]。

しかし、リモートペアプログラミングを行うことにより通常のペアプログラミングに比べて、ドライバやナビゲータがお互い相手の状況や状態を感じとることができなくなるためコミュニケーションがうまくいかなくなってしまう。この問題に対して、既存の研究[1][2]では、音声チャットや相手の状態(退席中など)のリアルタイムでお互いの情報をやり取りすることでコミュニケーション問題を解決しようとしている。

しかし、まだ問題がある。

1つ目は、利用者は自分の状況と相手から送られてくる情報を基に臨機応変に対応しなければならないことである。多くの情報の中から必要な情報を選び、作業を行いながら相手がどのような状況や状態に置かれていて、自分が何をすることができるのかをいつも考えて行動しなければならないため、難しい。

2つ目は、お互いに通知される情報には強制力がないので、必ず相手に伝わっているかは分からないことや伝わっていてもそれに適した行動をしてもらえないかは分からないことである。例えば、ドライバが間違った方向へプログラミングしていることにナビゲータが気づいてドライバの作業を止めようとした場合、通常のペアプログラミングでは、ドライバは隣にいるナビゲータの表情や雰囲気などを感じ取ることができる

め、ナビゲータの要求を受け入れてくれる。しかしリモートペアプログラミングでは、ドライバは離れているナビゲータの表情や雰囲気などを感じ取ることができないため、ナビゲータの要求を受け入れてくれるとは限らない。また、ドライバはプログラミングに夢中になってしまうことがあるため、ナビゲータからの情報がドライバに伝わっていないこともある。

これらの問題に対して、本稿では、ドライバとナビゲータの役割とその状況や状態を詳細に定義してシステムに管理させることで、システムが利用者の状態を把握してそれに合った機能を提供することによって解決する方法について検討する。

2. システム概要

まずリモートペアプログラミングにおけるドライバとナビゲータの状態を詳細に定義して、それら2つの直積によってペアの状態を定義するとともに、ペア状態間の遷移のきっかけとなる事象を洗い出す。

2.1. ドライバとナビゲータの状態の定義

以下の状態を洗い出した。

○開始時の共通状態

・役割分担

利用者の役割(ドライバ、ナビゲータ)を決める。

○ドライバ

・プログラミング(試し、修正案要求なし)

ナビゲータからメモ(試し、修正案)作成要求はまだ来ていない。

・プログラミング(試し、修正案要求あり)

ナビゲータからメモ作成要求が来たため、ドライバは作業がまとも次第、メモ作成へ移行する。

・デバッグ(試し、修正案要求なし)

ナビゲータからメモ(試し、修正案)作成要求はまだ来ていない。

・デバッグ(試し、修正案要求あり)

ナビゲータからメモ作成要求が来たため、ドライバは作業がまとも次第、メモ作成へ移行する。

・コードリーディングとアドバイス

ナビゲータが作成しているメモのコードリーディングとアドバイスを行う。

・作成したメモ(試し、修正案)の確認

ナビゲータが作成したメモを実行し、メモとその実行結果をお互いに確認し合う。

・プログラムと作成したメモ(試し、修正案)の比

A solution to resolve communication problems in remote pair programming environment

[†] SHUICHIRO ENDO, Graduate School of Informatics, Shizuoka University.

[‡] TSUYOSHI OHTA, Faculty of Informatics, Shizuoka University.

較、検討

ドライバのプログラムとナビゲータのメモの両方を表示して、比較や検討を行う。

○ナビゲータ

- ・コードリーディングとアドバイス
ドライバが作成しているプログラムのコードリーディングとアドバイスを行う。
- ・デバッグ
ドライバがデバッグをしているプログラムのコードリーディングとアドバイスを行う。またデバッグツールがある場合、この状態で利用する。
- ・交代依頼の確認、判断
ドライバからの交代依頼を確認し、判断する。
- ・メモ (試し、修正案) の作成
ナビゲータが試したいことや修正案 (コンパイルや実行できるメモ) を作成する。
- ・作成したメモ (試し、修正案) の確認
ナビゲータが作成したメモを実行し、メモとその実行結果をお互いに確認し合う。
- ・プログラムと作成したメモ (試し、修正案) の比較、検討
ドライバのプログラムとナビゲータのメモの両方を表示して、比較や検討を行う。

2.2.ペア状態の定義と遷移

2.1 にて定義したドライバとナビゲータの状態の直積を取り、それをペア状態とする。ただし、そのペア状態の中から、ドライバとナビゲータが共に単独で作業している状態は、ペアプログラミングにおける「両者が常に一緒に作業すること」の原則に反するので、除くことにする。

それぞれの状態間の遷移のきっかけとなる事象を洗い出した結果、状態数が 19 個の状態遷移図を作成することができた。図 1 に作成した状態遷移図の一部を示す。

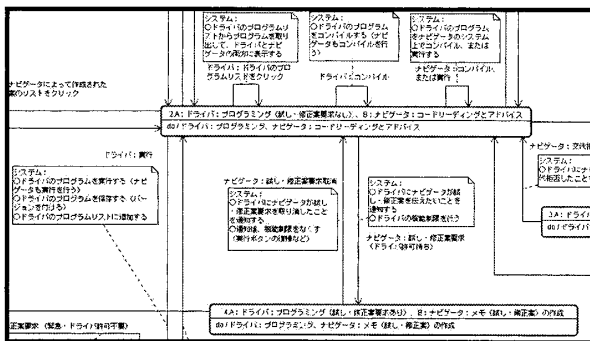


図 1: 作成した状態遷移図の一部

2.3.システム構築

作成した状態遷移図を基に、システムを構築した。

システム構築には、Java の Swing と JXSE[4]と呼ばれる P2P ネットワークを使用した。

図 2 に構築したシステムのドライバ側の画面を示す。

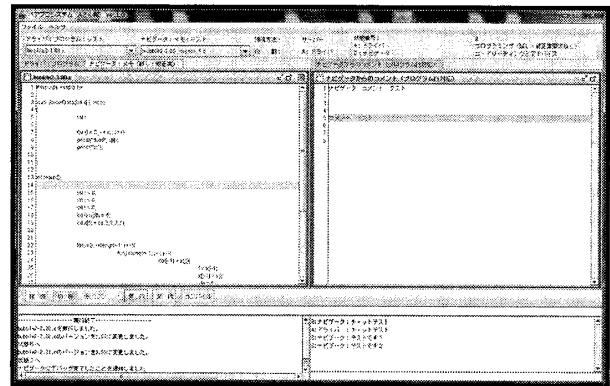


図 2: 構築したシステム (A 側: ドライバ)

今回構築したシステムは既存システム[3]とは違い、利用者の役割や作業を状態としてシステム側で管理し、それぞれの状態に合わせて、利用者が行える作業や機能、ボタンを提供している。また、ユーザがボタンの 1 つを押すことで次状態へと遷移する。右側にはナビゲータがコメントを書くことができるウィンドウを用意し、これとは別にあくまでもメモという位置づけとすることでナビゲータもプログラムを書くこと (メモ作成) ができるウィンドウも用意した。

3.おわりに

今回構築したシステムと既存システムとの比較実験と評価にあたり、それぞれのシステムに対して以下のような点からデータを収集する予定である。

- 自分がやりたいこと、また相手にそれが伝わっているか
- 相手がやりたいこと、また自分にそれが伝わっているか
- 意見の対立や相手の強行の対応具合
- 各システムにしかない機能に対する評価

参考文献

[1] 富田宣也, 片岡裕雅, 郷原麻衣子, 箱崎正洋, バトバートル, 長尾和彦: 「XP に基づいたプログラミング学習システムの評価」, 信学技報 ET 105(632) pp.121-124 (2006)

[2] 熱田智士: 「分散 XP 支援環境の構築」, 芝浦工大, <http://www.sayo.se.shibaura-it.ac.jp/SELHomepage/abstract/2003/p00005.pdf> (2010年1月13日)

[3] Xpairtise - Pair Programming for Eclipse <http://xpairtise.sourceforge.net/> (2010年1月13日)

[4] JXTA JXSE <https://jxta.dev.java.net/> (2010年1月13日)
<https://jxta-jxse.dev.java.net/> (2010年1月13日)