

ペアプログラミング演習のための 意味記述付きデータ構造図を用いた対話システム

古田 剛三[†] 立岩佑一郎[†] 山本 大介[†] 高橋 直久[†]

[†] 名古屋工業大学大学院 〒466-8555 愛知県名古屋市昭和区御器所町

E-mail: [†]furuta@moss.elcom.nitech.ac.jp, ^{††}{tateiwa,yamamoto.daisuke,naohisa}@nitech.ac.jp

あらまし 我々は分散型ペアプログラミング演習システム DIPES を開発し演習に適用している．DIPES では対話を文字列に加えて画像を共有して行う．しかし，現状の画像共有機能では，画像の作成に手間がかかる，注目データやデータ間の関係の記述が曖昧になる，複雑な画像の理解に時間がかかるという問題が生じる．本稿では，この問題を解決するためにペアプログラミング演習のための意味記述付きデータ構造図を用いた対話システムを提案し，その実現法を述べる．提案システムは意味記述付きデータ構造図生成機能，データ間関係記述機能，データ構造表現抽象化機能を有する．

キーワード e-learning, ペアプログラミング演習, 意味記述付きデータ構造図, 対話

A System for Supporting Conversations Based on Data Structure Diagrams with Semantic Descriptions in Pair Programming Exercises

Gozo FURUTA[†], Yuichiro TATEIWA[†], Daisuke YAMAMOTO[†], and Naohisa TAKAHASHI[†]

[†] Graduate School of Engineering, Nagoya Institute of Technology

Gokiso-cho, Showa-ku, Nagoya, Aichi 466-8555 Japan

E-mail: [†]furuta@moss.elcom.nitech.ac.jp, ^{††}{tateiwa,yamamoto.daisuke,naohisa}@nitech.ac.jp

Abstract We develop Distributed Pair Programming Exercise System(DIPES). Users converse with the pair user on DIPES with character strings and images. However there are problems in the existing picture share function: 1) Creation of a picture requires much time and effort. 2) Description of the attention data or the relation between the data becomes ambiguous. 3) Understanding of a complicated picture takes time. This paper proposes Conversation System Using Data Structure Diagram with Semantic Description for Pair Programming Exercise. The proposed system has functions: generation function of data structure diagram with semantic description, description function of relation between data, and abstraction function of data structure expression.

Key words e-learning, pair programming, Data Structure Diagram with Semantic Description, converse

1. はじめに

我々は分散型ペアプログラミング演習システム DIPES [1] を開発し，演習に適用している．DIPES ではネットワークを介して，ソースコードのリアルタイムな編集，チャットによる問題文に対する議論，アノテーションによるソースコード起点の議論を行う．

DIPES ではペアの対話は文字列のみが用いられる．そのためデータ構造やプログラムの流れなどの複雑な議論を円滑に行えないという問題が生じる．

解決策として画像を共有し対話に用いることが考えられる．我々はソースコードからデータ構造に基づいて編集可能な図

を作成し，DIPES 上で共有可能とするシステムを開発している [2]．このシステムにおいては以下の利点が得られる．

利点 1 図の生成にかかる手間を小さくできる．

利点 2 データ構造に基づき編集可能である．

しかし，上記システムには以下の問題点がある．

問題点 1 図の作成の際にユーザはシステムを用いてソースコードから図を生成し，生成した図を編集して対話に用いる図を完成させる．図の編集に時間がかかることで円滑な対話の妨げになる．また，類似画像を利用したい場合であっても図の再利用が難しい．

問題点 2 フリーハンドで描画した矢印などにより，注目データやデータ間の関係を記述しても，曖昧性があるため相手に意図

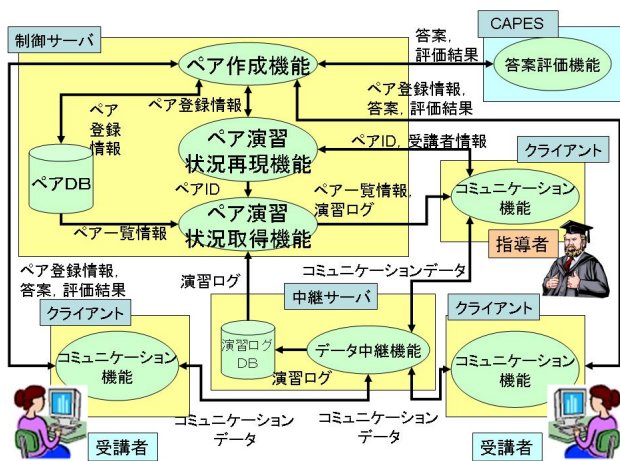


図 1 DIPES の構成図

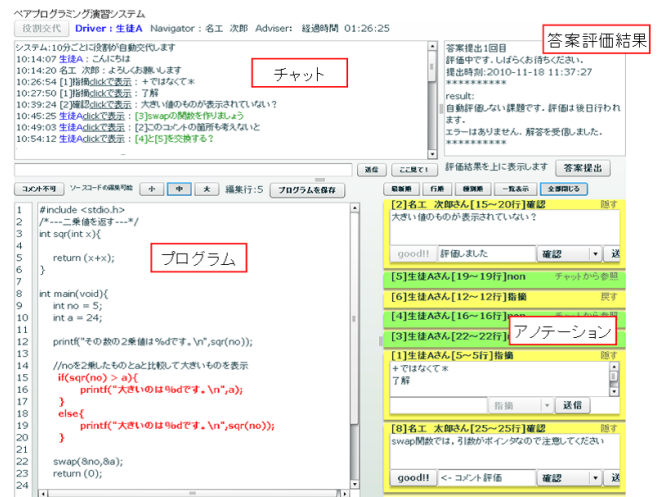


図 2 DIPES の実行画面

が正しく伝わらない場合がある．また，プログラムの動作を伝えるためにデータ間の関係を表す図を時系列順に複数作成する場合，すべての図に対してデータ間の関係を記述しなければならない．

問題点 3 大きな配列や構造体，構造体の配列，複雑な構造体などをそのまま描画すると図が複雑になり，図の理解に時間を要する．説明内容に応じて図を簡単化する必要がある．

本稿では，上記問題点に対する仕組みとして，意味記述付きデータ構造図を用いた対話システムを提案する．

2. 分散型ペアプログラミング演習システム

2.1 概要

分散型ペアプログラミング演習システム DIPES は受講者はネットワークを介して，ソースコードのリアルタイムな編集，チャットによる問題文に対する議論，アノテーションによるソースコード起点の議論を行う．さらに我々が開発，運用しているプログラミング演習支援システム CAPES [3] [4] [5] と連携し，答案の自動評価を行う．

DIPES の構成図を図 1 に示す．DIPES は制御サーバ，中継サーバ，クライアントシステムから構成される．また，CAPES と連携し，答案の自動評価を行う．各構成要素について述べる．

制御サーバではペアの作成を行う．受講者がログインした際にペアを作成し，ペアプログラミング演習の準備を行う機能を持つ．

中継サーバはクライアントシステムのコミュニケーションデータをペアの相手に対して中継する．

クライアントシステムではソースコードの変更，チャットおよびアノテーション機能を提供する．受講者はクライアントシステムを操作して DIPES を利用する．

ペアで作成したプログラムは CAPES に対して提出され自動評価される．自動評価の結果はクライアントシステム上で表示される．

2.2 クライアントシステム

本論文で提案しているシステムは DIPES のクライアントシステム上で動作する．よってクライアントシステムについて詳

しく述べる．

クライアントシステムの実行画面を図 2 に示す．クライアントシステムはプログラム領域，アノテーション領域，チャット領域，答案評価結果領域から構成される．

プログラム領域はペアでリアルタイムなソースコードの共有がされている．Driver はこの領域においてソースコードを編集する．

アノテーション領域は，ソースコードに対するアノテーションを表示する．アノテーションは編集されるソースコードの変化に対応する協調的アノテーションを提案している．アノテーションは議論の内容毎に作成され，対応するソースコードはハイライト表示される．ソースコードのある部分に対して議論を行いたい場合に用いられる．

チャット領域ではペア間での会話を行う．また，アノテーションで行われた議論も表示される．よってチャット領域ではペアの議論が時系列順に表示される．

答案評価結果領域では，答案を CAPES へ提出した際の評価結果を表示する領域である．

STEP1 演習項目の理解

まず，受講者は演習項目を理解することを目指す．演習項目の理解のために以下を行う．

- (1) 講義資料を読む
- (2) 講義を聴く
- (3) 確認テスト

講義資料には演習項目に対する解説が記されている．受講者は演習中に講義資料を参考にして演習を進める．講義は指導者が複数名の受講者を対象として，講義資料を基に演習項目の解説を行う．確認テストでは講義終了後に演習項目の理解の確認のために課題を解き，指導者へ提出する．

STEP2 演習問題を解く

受講者は演習項目に対応したプログラムの作成を行う．プログラミングは DIPES 上で行われる．プログラムの作成には以下の手順がある．

- (1) プログラムの設計
- (2) コーディング
- (3) デバッグ

プログラムの設計では、コーディング前にペアで対話を行い、プログラムの設計について確認を行う。コーディングでは Driver と Navigator に分かれてペアプログラミングを行う。デバッグではコーディングと同じく Driver と Navigator に別れ、プログラムの問題点を修正していく。

3. 提案システムの特徴

提案システムは以下の特徴を持つ。

特徴 1 意味記述付きデータ構造図生成機能

配列および構造体を記述したプログラムから、データ構造とその意味を表すメタデータからなる図を生成する。生成した図はシステム上でメタデータを用いることでデータ構造に基づき編集が可能である。また、メタデータ付きの画像ファイルとして出力ができる。この機能により、DIPES の通信機能によらないデータ構造図の受け渡しが可能となる。よって、あらかじめ指導者が演習で使用されるであろう図を生成しておき、受講者が再利用できるようになる。これにより図の変更と再利用を容易にし、問題点 1 の解決法となる。

特徴 2 データ間関係記述機能

ポインタ、インデクス、代入などプログラム中に記述されたデータの関係、および、閲覧時の視点移動を伴うデータ間の関係を、関係の種類毎に異なる矢印アイコンにより記述し、図に埋め込む。これにより、図を用いた対話における曖昧性を減少させることが可能になる。また、プログラムの実行時のデータ間の関係の変化を反映した図を自動的に生成可能となる。この機能により問題点 2 を解決する。

特徴 3 データ構造表現抽象化機能

データ構造図を抽象化して描画する機能を実現する。これにより、受け手の状況に応じてデータ構造図の表現形式を変えることでもんだ点 3 を解決する。例として、構造体の配列の図を考える。図の受け手に対してプログラム作成の際の説明では構造体が複数並ぶ図を表示する。一方、図の受け手に対してアルゴリズムを説明する際には構造体の要素をひとまとめにし、図を簡略化することで理解しやすくする。例を図 3 に示す。ここでは構造体の配列を表している。図 3 の上のデータ構造図は構造体を複数並ぶ図となっており、図 3 の下の図は表のような図となっている。プログラム作成時は上の図、アルゴリズムの説明時には下の図を用いる。

4. 提案システムの実現法

4.1 意味記述付きデータ構造図生成機能

意味記述付きデータ構造図はメタデータと表現形式からなる。メタデータはデータ構造の構造を表す。表現形式はデータ構造をどのように表現するかを定義する。

ソースコードを正規表現により解析し、変数、配列および構造体の定義と宣言を取得する。取得した情報から、データ構造

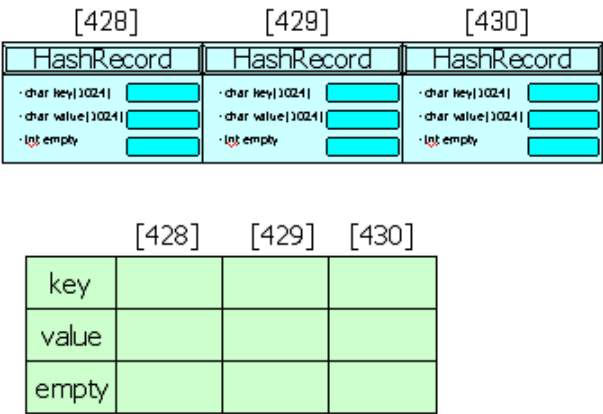


図 3 データ構造表現抽象化機能の例

表 1 メタデータの例

型	struct rect
識別子	r1
要素 1 型	int
要素 1 識別子	width
要素 1 値	10
要素 2 型	int
...	...

を表すメタデータを生成する。メタデータにはデータ構造の型、識別子名、値、元となるソースコードがある。構造体の場合の例を表 1 に示す。生成したメタデータと表現形式に基づきデータ構造図を生成する。

表現形式はメタデータをどのようにデータ構造図として表すかを規定したものである。表現形式はデータ構造図の抽象度によって変わる。

また、メタデータと表現形式を XML 形式で記述し、画像ファイルに埋め込むことによって、データ構造図を画像ファイルとして出力する。出力した画像ファイルを本機能に入力すると、画像ファイル内の XML 形式で記述されたメタデータと表現形式を読み取り、データ構造図を復元する。この機能により DIPES の通信機能を用いないデータ構造図の受け渡しが可能となる。

4.2 データ間関係記述機能

データ構造図の要素に対して一意に定めるアドレスを持たせる。データ間の関係の始点と終点をアドレスを用いて関連付け、矢印などを描画することで実現する。データ間の関係を表す図の例を表 2 に示す。

ポインタはポインタ変数を始点とし、ポインタの指す先のアドレスを終点とする。インデクスは配列のインデクスを表した数値型の変数を始点とし、終点は配列のインデクス番目とする。代入は始点を代入元の変数とし、代入先の変数を終点とする。視点の移動は始点の移動元を始点とし、移動先を終点とする。注目点は始点を持たず、注目すべき点を終点とする。

また、プログラム中にその時点でのデータ構造図を生成する Snapshot 関数を記入することによりプログラムの実行時の

表 2 データ間の関係を表す図

関係の種類	関係を表す図
ポインタ	●——→
インデックス	●-----→
代入	●——→
視点の移動	→
注目点	▷

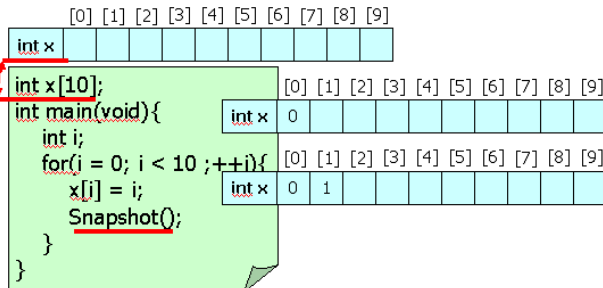


図 4 Snapshot 関数

データ間の関係の変化を反映した図を自動的に生成する。まず、プログラム実行時にデータ構造図とプログラム中の変数の対応付けを行う。データ構造図のアドレスとプログラム実行時の変数のアドレスを対応付けることでデータ間の関係を更新し、意味記述付きデータ構造図生成機能によりデータ構造図を生成する。さらに、アドレスから変数の値を取得し、変数の値の変化を図に反映する。例を図 4 に示す。例では `int x[10]` という変数とデータ構造図を対応づけている。Snapshot 関数が呼ばれるたびに最初のデータ構造図と変数の値を基に新しいデータ構造図を生成している。

4.3 データ構造表現抽象化機能

データ構造図のメタデータをもとに表示形式を変更する。変更する表示形式はメタデータの情報の表示・非表示、表示位置の変更で行う。また、配列データを 1 つの変数として表示および変更を可能とする。本機能はユーザの目的に合わせた要素の表示・非表示の組み合わせのプリセットを提供する。ユーザは提示されたプリセットから目的に合わせて選択し、データ構造図の表現を変更する。

5. 提案システムを用いた演習の流れ

提案システムを用いた演習は以下のステップで行われる。

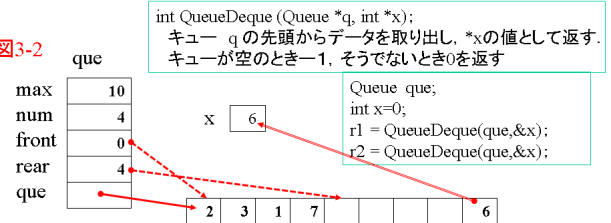
ステップ 1 講義資料作成

提案機能を用いて講義資料を作成する。講義資料の例を図 5 に示す。データ構造画像を生成し、用途に合わせて抽象度を変更する。データ構造画像間の関係および注目点を指定する。最後に値を入力する。もしくは実行時データ構造画像出力機能を用いて画像を生成する。生成したデータ構造画像を画像ファイルとして出力し、講義資料に用いる。また、画像を用いた課題を作成する。画像を用いた課題はデータ構造画像を編集し提出する課題である。

ステップ 2 講義

QueueDeque(que,&x)の実行結果

図 3-2



問3

- ① 図3-1のキューqueに対して、`r1 = QueueDeque(que, &x)`を実行した後のqueのデータ構造と変数x値を表す図を作成しなさい。
- ② `QueueDeque(que, &x)`をもう一度実行した後のqueのデータ構造とxの値を表す図を作成しなさい。
- ③ 関数`QueueDeque(Queue *q, int *x)`のプログラムを作りなさい。

図 5 講義資料の例

作成した講義資料を用いて講義を行う。講義は指導者が多数の受講者に対して説明を行う。

ステップ 3 確認テスト

講義資料に掲載されている画像を用いた確認テストに取り組む。この確認テストは DIPES 上でペアで相談しながら取り組む。この確認テストの狙いは講義の内容の理解を深めることと、ペアでのコミュニケーションを促すことである。確認テストの提出は画像ファイル入出力機能により出力された画像ファイルで行う。

ステップ 4 ペアプログラミング

DIPES を用いてペアプログラミングを行う。講義の内容に沿ったプログラムを作成する。

6. プロトタイプシステムの実装

提案システムの意味記述付きデータ構造図生成機能およびデータ構造表現抽象化機能の一部をプロトタイプシステムとして実装した。C 言語を対象とし、プロトタイプシステムには Flex [6] を使用した。プロトタイプシステムを図 6 に示す。図 6 の A にソースコードを入力する。B ではソースコードを解析し、生成可能なデータ構造図の一覧である。C はデータ構造表現抽象化機能の指定箇所である。D の読み込みボタンは意味記述付きデータ構造図生成機能で生成した画像ファイルを読み込むために使用する。

6.1 意味記述付きデータ構造図生成機能

利用者は A にソースコードを入力し、「ソース解析」ボタンを押す。システムはソースコードを解析し、生成可能なデータ構造図の一覧を B に表示する。利用者は B から生成するデータ構造図と、C から表示方法を選択し、「生成」ボタンを押すと E のデータ構造図が生成される。

データ構造図はデータ構造に基づく編集が可能である。プロトタイプシステムでは、値の入力、配列の要素数の変更、構造体の要素の削除、データ構造図の見出しの文字列の変更が可能である。

また、生成したデータ構造図を右クリックし、「保存」をメニューから選択すると、メタデータを含む画像ファイルを出力

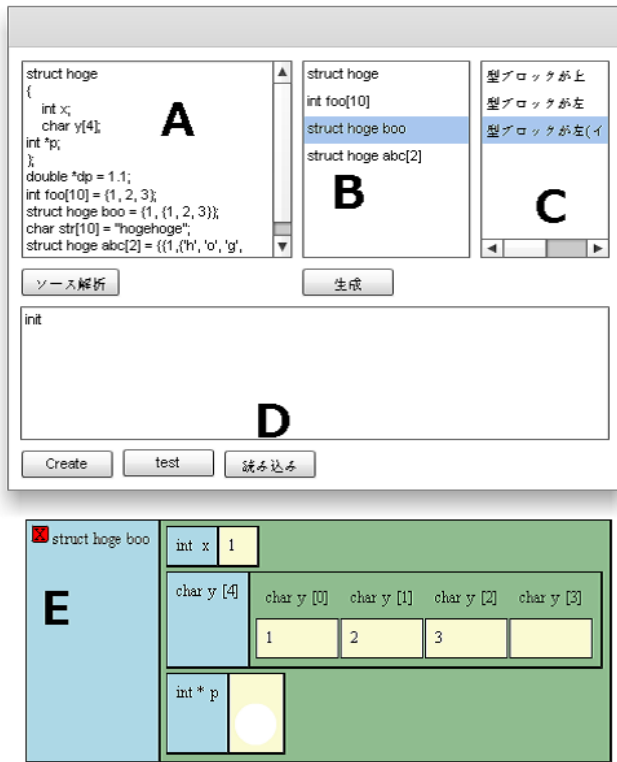


図 6 プロトタイプシステム

する．プロトタイプシステムでは JPEG 形式の画像ファイルとして出力する．JPEG ファイルのコメント部にメタデータと表現形式を書き込んでいる．

6.2 データ構造表現抽象化機能

プロトタイプシステムでは，データ構造表現抽象化機能として，以下の項目を実装した．

- 要素の並び方の変更
- 文字列を 1 変数として表示

要素の並び方の変更は図 6 の C で選択する．要素名と値を縦方向に並べるか横方向に並べるかを選択できる．また，配列のインデックスのみ縦方向に並べ，残りを横方向に並べるようにしている．

文字列を 1 変数として表示する機能は，文字の配列で表される文字列を，1 つの文字列変数として表示するものである．例えば，{'A','B','C','\0'} という文字の変数を"ABC"と表示する．図 7 に実行例を示す．文字列操作の課題では文字の変数を使用し，それ以外では文字列変数として表示することで理解のしやすさを高めている．

7. 評価実験

提案システムが演習に適用できるかどうかを検証する．名古屋工業大学第 1 部 2012 年度プログラミング II の演習 12 単元の講義資料に用いられているデータ構造図を抽出し，提案システムおよびプロトタイプシステムで適用可能であるか確認した．適用可能とは，提案機能を用いれば講義資料と同じ図を生成できることをいう．また，提案システムで生成可能な図に対しては，提案機能の内いずれの機能を用いれば適用可能かを検

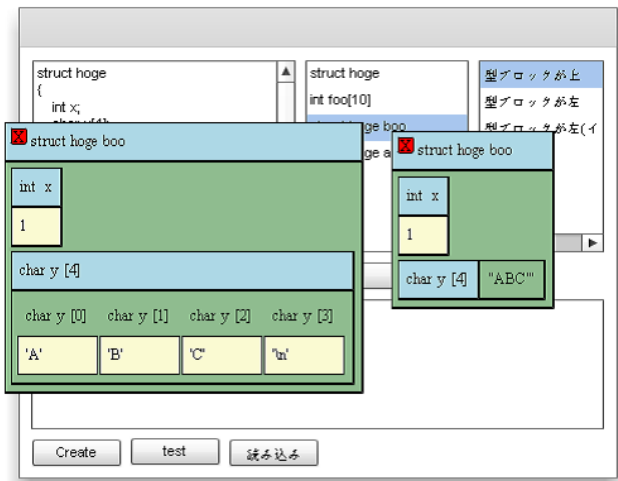


図 7 文字列を 1 変数として表示

表 3 提案システムおよびプロトタイプシステムの評価結果

	適用可能数	適用可能率
提案システム	108/135	80.0%
プロトタイプシステム	23/135	17.0%

5-10

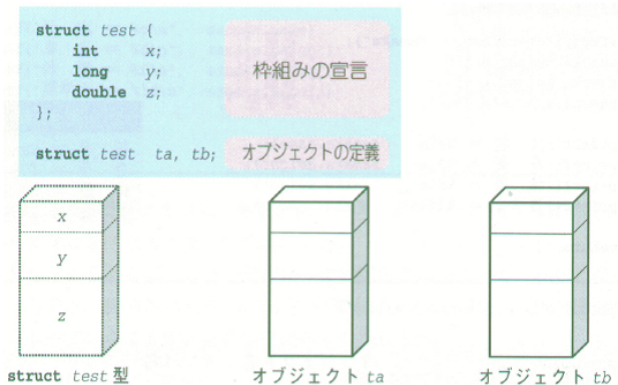


Fig.12-4 構造体の宣言とオブジェクトの定義

図 8 提案システムが適用できない例 サイズを反映した図

証した．

評価結果を表 3 に示す．提案システムは講義資料の図の内 80%に適用可能であった．また，プロトタイプシステムでは 17%が適用可能であった．プロトタイプシステムは提案機能の内，データ間関係記述機能とデータ構造表現抽象化機能が実装されていないため，適用可能率が低くなったと考えられる．

提案システムで適用不可能な図の例としてデータ構造のメモリ上のサイズを考慮したデータ構造図があった．

提案システムが適用できなかった理由として，データ構造のサイズをデータ構造図のサイズに反映させる表示をすることが提案システムではできない．例を図 8 に示す．この例ではサイズが異なるメンバを持つ構造体の図を表示している．このように，サイズの大きなメンバを大きく表示することが提案システムはできない．

8. おわりに

本稿ではペアプログラミング演習のための意味記述付きデータ構造図を用いた対話システムを提案した．また，提案した実現法を基に一部機能のプロトタイプを実装した．さらに提案システムとプロトタイプシステムの評価を名古屋工業大学の演習の講義資料をもとに行った．

今後の課題として，提案機能の DIPES への組み込みおよび，実際の演習での評価を行い，提案機能の有用性を検証することが挙げられる．

文 献

- [1] 青木一浩，立岩佑一郎，山本大介，高橋直久，ソースコードに関する協調的な対話機能を有する分散型ペアプログラミング演習システムの実現，第3回データ工学と情報マネジメントに関するフォーラム (DEIM2011), F8-5 (2011)．
- [2] 古田剛三，立岩佑一郎，山本大介，高橋直久，分散型ペアプログラミング演習システム DIPES のためのデータ構造スタンプによる画像コミュニケーション機能の実現，情報処理学会第74回全国大会，1ZE-4，2012/3.
- [3] 中島秀樹，宮地恵佑，高橋直久，”プログラミング演習支援システム CAPES のための答案評価機構の実現”，情報処理学会研究報告，2006-CE-83(18)，pp127-134，2006.
- [4] 中島秀樹，高橋直久，プログラミング学習のための QA サイクル受講者の習得度に応じた問題自動提示メカニズム，電子情報通信学会論文誌，VOL.J88-D-I，NO.2，2005
- [5] 中島秀樹，高橋直久，細川宜秀，受講者の状況に応じた個別学習機能を有する多言語対応プログラミング学習システムの実現，電気情報通信学会第15回データ工学ワークショップ (DEWS2004)，2004
- [6] オープンソースフレームワーク ”Adobe Flex”，<http://www.adobe.com/jp/products/flex/>