

iComb: um sistema para o ensino e aprendizagem de combinatória em ambiente Web

Alexandre Luís Kandrát Eisenmann¹, Leônidas de Oliveira Brandão¹

¹Instituto de Matemática e Estatística – Universidade de São Paulo (USP)

***Abstract.** In this work, we present the software **iComb - Interactive Combinatorics on Internet**, a new software that can be used to teach and learning combinatorics through the World Wide Web. The iComb is based on the Combien? experience, a software developed by Université Pierre et Marie Curie - LIP6. We can easily integrate the iComb software in a web-based learning management system (LMS), as we did in two LMS, one of them was a version of Moodle system. Moreover, the iComb can be easily added to a personal web sites, social networks systems and blogs using the copy-and-past pattern resulting in a powerful collaborative tool to the conceptual movement named as Web 2.0.*

***Resumo.** Neste trabalho apresentamos um novo sistema de apoio ao ensino/aprendizagem via Web, o iComb. Também apresentamos alguns experimentos didáticos com a atual proposta do iComb. Este sistema é baseado na experiência do Combien?, desenvolvido pela Université Pierre et Marie Curie, que tem bons recursos para auxílio à resolução de exercícios ligados a análise combinatória, mas apresenta limitações quanto a incorporação de novos exercícios e principalmente apresenta restrições para seu uso via Web. Uma inovação advinda deste projeto é que o iComb pode ser integrado a sistemas gerenciadores de cursos, como o Moodle. Além disso, o iComb pode ser incorporado a páginas pessoais, redes sociais e blogs, na forma de um “widget”.*

1. Introdução

A Análise Combinatória é a parte da matemática que analisa estruturas e relações discretas. Um dos tipos de problemas que ocorre com frequência neste domínio é contar ou classificar os subconjuntos de um conjunto finito que satisfazem certas condições dadas sem que seja necessário enumerar seus elementos [Morgado et al. 1991].

Embora a Análise Combinatória possua técnicas gerais que permitem atacar certos tipos de problemas, a solução de um problema combinatório exige quase sempre engenhosidade e a compreensão plena da situação descrita pelo problema. Por outro lado, se a aprendizagem destes conceitos se faz de maneira mecânica, padronizada, cria-se a impressão de que a Análise Combinatória é somente um jogo de fórmulas complicadas [Morgado et al. 1991].

Giroire et al. (2006) compartilham deste ponto de vista ao afirmarem que problemas de contagem e combinatória em geral não são resolvidos através de um método dedutivo, onde a partir de um conjunto de axiomas aplica-se regras de inferência para a obtenção de novos fatos até a resolução do problema. Segundo estes autores, a reação dos alunos frente à resolução de problemas de combinatória reflete uma dificuldade de representação, como ilustrado nas seguintes frases relatadas por eles¹: “Não consigo re-

¹Frases traduzidas do inglês pelos autores.

presentar o problema”, “Não sei por onde começar”, “Eu entendo a solução dada pelo professor, mas não entendo porque a minha está errada”, “Eu propus uma solução mas não sei dizer se está certa ou errada” [Giroire et al. 2006].

Deste modo, é necessário encontrar meios de apresentar o assunto de forma a evitar que a maioria dos alunos entenda a disciplina como um receituário de fórmulas que só resolvem casos particulares e que apenas os alunos muito criativos sejam capazes de encontrar bons modelos de solução. Uma forma de se aproximar deste objetivo é fornecer recursos para que os alunos “enxerguem” o problema e “visualizem” as respostas, proporcionando significado à experiência realizada como recomendado pelos Parâmetros Curriculares Nacionais², o que pode ser conseguido com o uso de software adequado.

Além de ajudar na visualização, as Tecnologias da Informação e da Comunicação (TIC) podem trazer outros benefícios para o ensino e a aprendizagem de matemática. Particularmente, o uso de sistemas interativos e a infra-estrutura da Internet tem se mostrado muito útil para o ensino/aprendizagem de Geometria, com o uso do que se convencionou denominar Geometria Interativa (ou Dinâmica) [Laborder e Bellemain 1997, Laborder e Bellemain 1997]. Mais recentemente (1993) teve início o projeto *Combien?*, da Université Pierre et Marie Curie - LIP6 (Paris, França), visando fornecer um modelo para ensino/aprendizagem de Combinatória [Le Calvez et al. 2003, Giroire et al. 2006, Tisseau et al. 2000]. Entretanto este sistema não pode ser utilizado diretamente em ambiente Web.

Este trabalho foi motivado principalmente pelos resultados do grupo *Combien?* e do uso de sistemas de Geometria Dinâmica integrados à ambientes Web. Desta motivação resultou o desenvolvimento do sistema iComb, destinado ao ensino/aprendizagem de Combinatória na Internet.

A contribuição do trabalho para a Informática na Educação é prover um sistema adequado ao uso na Web capaz de auxiliar no ensino e aprendizagem de Combinatória. Como será visto na Seção 4, o iComb é mais do que uma reimplementação do *Combien?*. O iComb possui uma arquitetura extensível (*arquitetura de plugins*) onde novos “temas” de exercícios podem ser incorporados; mecanismo de internacionalização; repositório dinâmico de exercícios e um novo modelo de imagens como apresentado na Figura 3. Além disso, a fundamentação teórica foi desenvolvida de forma independente, e bastante diferente, do *Combien?* proporcionando um algoritmo rápido de avaliação automática de erros.

Na próxima Seção será discutida a importância do retorno (*feedback*) rápido para as atividades submetidas pelos alunos em ambientes de aprendizagem Web. Na Seção 3 será apresentado o *Combien?* e a forma como os exercícios são nele resolvidos. Na Seção 5 serão apresentados alguns experimentos realizados com as primeiras versões do iComb. Na Seção 6 estão as conclusões deste trabalho e indicações de alguns trabalhos futuros.

²“A insatisfação revela que há problemas a serem enfrentados, tais como a necessidade de reverter um ensino centrado em procedimentos mecânicos, desprovidos de significados para o aluno. Há urgência em reformular objetivos, rever conteúdos e buscar metodologias compatíveis com a formação que hoje a sociedade reclama.” [BRASIL 1997, pg. 15]

2. Respostas de avaliações em ambientes de aprendizagem na Web

Como observado por Giroie et al. (2006), geralmente no processo de aprendizagem, os erros cometidos pelos alunos são importantes pois constituem situações particularmente favoráveis para o aprendizado. Em sala de aula tradicional³, os alunos resolvem exercícios, mas têm que esperar por longos períodos de tempo para terem acesso a modelos de resposta. Quando a solução finalmente lhes é fornecida, eles já esqueceram o contexto do problema e o melhor momento para apresentar correções passara [Giroire et al. 2006].

Vários outros trabalhos apontam para a importância de **retorno** (*feedback*) rápido para as respostas de alunos, principalmente quando se trata de atividades via Web. Por exemplo, Hara e Kling (1999) e Hentea et al. (2003) afirmam que uma das principais frustrações de alunos envolvidos em cursos Web (incluindo aqueles com parte do curso sendo presencial) é a falta de retorno rápido sobre as tarefas enviadas pelos alunos. Já Moura et al. (2007) relata outro ponto positivo da resposta rápida, aliada a uma maior personalização proporcionada por um SGC, que é aumentar o envolvimento dos alunos com as atividades solicitadas.

Os modelos implementados no *Combien?*, assim como no iComb, apresentam funcionalidades para avaliação automática, proporcionando assim retorno imediato às respostas elaboradas pelos alunos. Ambos os sistemas podem, a cada etapa da solução, verificar se as ações até então realizadas pelo aluno podem ou não levar a um resultado correto. Desta maneira, o sistema poderá apontar rapidamente ao aluno um erro no desenvolvimento de sua solução. Alternativamente, o sistema armazena o erro para futura notificação. Como observado pelos autores do *Combien?*, a correção imediata do erro aumenta em muito a compreensão do aluno sobre o problema, uma vez que está ligado ao último raciocínio utilizado [Giroire et al. 2006].

3. Combien? e solução de problemas em estágios

A proposta do grupo *Combien?* da *Université Pierre et Marie Curie, LIP6*, foi construir um sistema pedagógico para ajudar alunos do sistema educacional francês a aprender combinatória utilizando linguagem matemática. Segundo seus autores, a interface do *Combien?* adota uma linguagem próxima àquela que eles espontaneamente tendem a utilizar na formulação dos problemas e que ao mesmo tempo, permite uma argumentação matemática rigorosa [Le Calvez et al. 2003, Giroire et al. 2006, Tisseau et al. 2000].

Em cada exercício no *Combien?*, o aluno deve determinar quantos possíveis elementos existem com determinada propriedade. Para ressaltar esta meta, os autores adotaram o termo francês correspondente a “Quanto?” como nome de seu sistema, *Combien?*. Inicialmente, o sistema apresenta ao aluno um texto com a descrição do problema e para responder à pergunta do tipo “quantos elementos”, o aluno deverá seguir alguns passos envolvendo ações, modelagem e raciocínio [Le Calvez et al. 2003, Giroire et al. 2006, Tisseau et al. 2000].

A fundamentação matemática subjacente ao método implementado no sistema foi chamada de *método construtivo*. A idéia é permitir que o aluno, raciocinando a partir das opções disponíveis no sistema, descreva um algoritmo de enumeração que seja solução

³Modelo presencial, sem recursos computacionais.

do problema. Segundo os autores do *Combien?*, essa é uma forma eficiente de resolver o problema uma vez que não é necessário “rodar” o algoritmo, apenas descrevê-lo criteriosamente [Le Calvez et al. 2003, Giroire et al. 2006, Tisseau et al. 2000].

A base de exercícios do *Combien?* é fixa, estando codificada no próprio sistema, que é um aplicativo desenvolvido em *SmallTalk*⁴. Todos os exercícios utilizam um único tipo de *universo*, que são as *cartas de baralho*.

Os exercícios solicitam do aprendiz que este estime quantos conjuntos com determinadas propriedades podem ser formados. Um exemplo típico de problema no *Combien?* é apresentado no quadro 1. Nesse problema o universo considerado é “Um baralho de 32 cartas”, o tamanho dos conjuntos de interesse é $N=5$ e as restrições do exemplo são “duas cartas de Copas” e “duas cartas de Espadas”.

Com um baralho de 32 cartas, quantas mãos de 5 cartas é possível formar com exatamente duas cartas de Copas e exatamente duas cartas de Espadas?

Quadro 1: Exemplo típico de problema no *Combien?*

Uma vez apresentado o problema, o sistema desafia o aluno a encontrar a cardinalidade do conjunto resposta para o problema, utilizando para isso a interface que o incentiva a quebrar a solução em *estágios*. Ao final o aluno deve definir como compor estes estágios, na forma de soma ou produto, para obter a resposta.

Um *estágio* possui uma estrutura bem definida, sendo composto por uma condição, que representa uma restrição sobre o conjunto universo, e um valor numérico, que representa o número de elementos que o aluno acredita atender esta restrição. Por exemplo, no problema do quadro 1, um dos estágios para resolvê-lo poderia ser “Eu quero [2] elementos que verifiquem [uma propriedade]: [naipe] [é] [Copas]”.

A Figura 1 apresenta a interface do *Combien?* para a resolução de um outro problema fazendo o uso de três estágios.

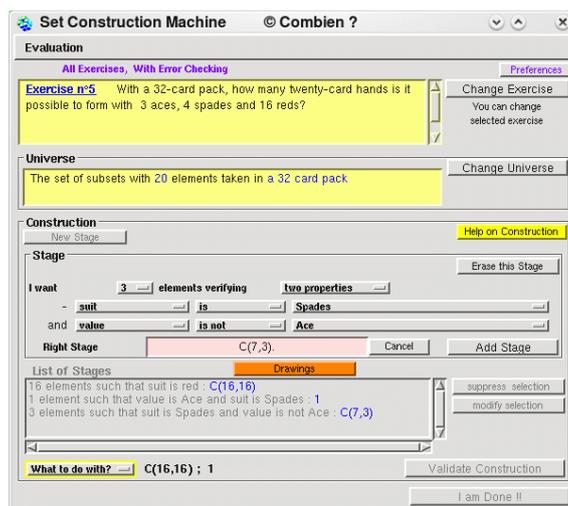


Figura 1. Interface *Combien?* com exercício e parte de uma resposta.

Entretanto, o *Combien?* apresenta duas restrições que dificultam seu uso em ambientes Web para ensino/aprendizagem. A primeira é por ser aplicativo executável, devendo

⁴O *SmallTalk* é uma linguagem de programação baseada em listas, vide [VW 2009].

portanto “rodar” localmente na máquina do cliente⁵, e a segunda por dispor de uma base fixa de exercícios, impedindo que grupos de professores desenvolvam novas atividades específicas para suas turmas. O desenvolvimento do iComb foi motivado principalmente para vencer estas duas restrições, como esposto a seguir.

4. iComb: modelo proposto

O iComb nasceu do esforço dos autores para incorporar de modo transparente as possibilidades didáticas do *Combien?* em cursos apoiados na Web. Deste modo, optou-se por adotar a linguagem Java⁶, o que viabilizou a utilização do iComb em páginas Web abertas ou integrado a SGC.

O iComb preserva características importantes do *Combien?*, tais como a detecção automática de erros e o registro do histórico de utilização (*log*). O iComb ainda não dispõe de todos os tipos de exercícios aceitos no *Combien?*, entretanto o primeiro estende o segundo em aspectos importantes para seu uso na Web: como citado, pode ser utilizado diretamente em ambientes Web; sua estrutura prevê internacionalização; é possível incorporar novos universos (além do baralho do *Combien?*); e possibilita a incorporação de novos exercícios, potencializando seu uso em repositórios de conteúdos didáticos digitais.

4.1. Arquitetura de Plugins

Segundo os Parâmetros Curriculares Nacionais, o significado da Matemática para o aluno resulta das conexões que ele estabelece entre ela e as demais disciplinas, entre ela e seu cotidiano e das conexões que ele estabelece entre os diferentes temas matemáticos [BRASIL 1997, pg. 19]. Uma das novidades introduzidas no iComb vai nesta direção de facilitar a aproximação do assunto no iComb com o cotidiano do aluno, por possibilitar a incorporação de novos universos, na forma de *plugin* garantindo um ambiente flexível para a inclusão de “temas” diferentes para um determinado conjunto de exercícios. Além disso o iComb apresenta uma interface de apresentação dos universos existentes, o que facilita a visualização por parte dos alunos para suas soluções.

Para incorporar um novo universo ao iComb, basta gerar um arquivo JAR⁷ contendo um descritor XML⁸ que defina o universo e um diretório com as imagens de todos os elementos do universo.

Atualmente o iComb dispõe de dois universos, *baralhos* e *times de futebol*. No primeiro, cada carta é representada por uma imagem da mesma e no segundo cada elemento é representado por uma camisa de time de futebol (poderia-se substituir estas por imagens de jogadores reais). Uma carta de baralho possui atributos como naipe e valor, já um jogador possui atributos como time e número.

⁵Para um funcionamento adequado de cursos via Web, é necessário um sistema do tipo cliente/servidor, que será denominado aqui como *Sistema Gerenciador de Cursos (SAW)*.

⁶O Java é uma linguagem aberta, proposta por um grupo de desenvolvedores da Sun Microsystem em 1995, que surgiu junto com a Web e pode ser utilizado dentro de navegadores, na forma de *applets*. Ela é hoje uma das linguagens mais utilizadas em ambientes Web.

⁷JAR (Java ARchive) é um arquivo compactado em formato *ZIP* que contém, tipicamente, classes Java e recursos relacionados.

⁸XML (eXtensible Markup Language) é, de forma geral, uma linguagem de marcação capaz de descrever diversos tipos de dados.

Existem quatro naipes de baralho, de maneira análoga existem 4 times de futebol (azul, verde, vermelho e amarelo). As cartas valete, dama, rei e ás são todas figuras. Os jogadores número 10 e 11 são atacantes enquanto o jogador número 1 é o goleiro. Assim, cada universo é definido com suas peculiaridades.

A arquitetura de *plugins* permite que exercícios análogos, porém de diferentes universos, sejam propostos aos alunos sugerindo a existência de uma estrutura geral e abstrata para os problemas, da mesma forma que uma solução concreta pode ser decomposta e compreendida utilizando-se conceitos gerais e abstratos como combinação, arranjo e permutação.

A Figura 2 apresenta a tela do iComb. As imagens apresentadas na tela durante a execução do exercício foram definidas no *plugin* baralho. A Figura 3 apresenta as imagens definidas para o *plugin* futebol.

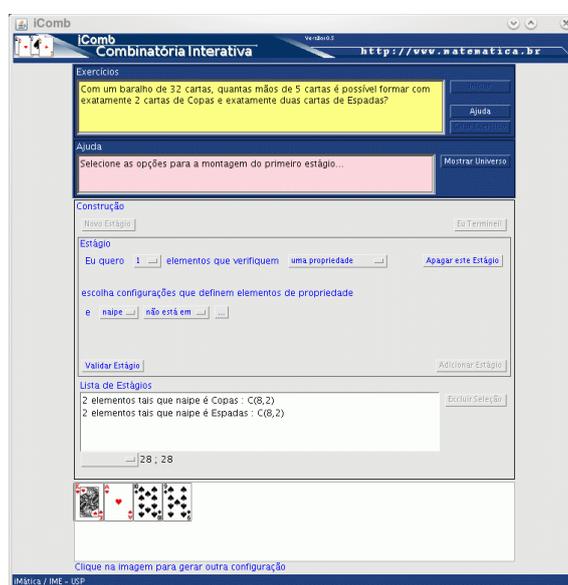


Figura 2. Tela do iComb apresentando um exercício com o *plugin* baralho.



Figura 3. Destaque para painel de imagens de um exercício com o *plugin* futebol.

4.2. Repositório de Exercícios

O iComb foi projetado de modo a permitir seu uso na construção de repositórios dinâmicos de exercícios. Assim, supondo sua utilização integrada a um sistema gerenciador de cursos (SGC), um professor poderá incluir novos exercícios à base de dados de exercícios⁹. Viabilizando deste modo a ampliação desta base de um ano para outro

⁹O iComb possui uma interface preparada para a composição de um novo exercício, não sendo necessário a definição do mesmo em arquivos XML. Porém o SGC deverá estar preparado para receber a chamada via protocolo HTTP.

ou por meio de troca de exercícios entre professores. O ideal é que o SGC disponha destes recursos para viabilizar o repositório de conteúdos digitais para comunidades de professores.

4.3. Fundamentação Teórica e Avaliação Automática

Um mecanismo de detecção automática de erros é um sub-sistema de software capaz de detectar, sem interferências externas, erros em ações realizadas pelo usuário. Geralmente este tipo de mecanismo é considerado melhor quando fornece informações sobre o erro cometido pelo usuário, embora existam situações nas quais o professor pode não desejar tal facilidade, como é o caso quando deseja-se apenas medir conhecimentos adquiridos pelo aluno. Em situações usuais de aprendizagem, principalmente na Web como aponta [Bridge e Appleyard 2008], apresentar retorno rápido às atividades dos alunos ajuda em seu aprendizado. Assim, quando o modelo de avaliação implementado permitir, é interessante não apenas avisar ao aluno se sua atividade está ou não correta, mas eventualmente intervir ao primeiro sinal de que sua tentativa de solução não poderá resultar numa resposta correta.

Nessa Seção será analisada a natureza e estrutura dos problemas propostos pelo iComb, discutindo-se até que ponto é possível a criação de um algoritmo de detecção de erro neste modelo. O objetivo será definir uma classe de problemas e um algoritmo que seja correto para esta classe. Neste contexto, será dito que um *algoritmo é correto para uma classe de problemas* quando, para qualquer problema da classe e qualquer que seja a proposta de solução apresentada, o algoritmo consegue detectar se esta é correta ou não.

Todos os problemas propostos para a versão atual iComb tem um enunciado que respeita sempre a mesma estrutura: “Quantos conjuntos de N objetos é possível formar respeitando as restrições ...”. Por exemplo, se o universo considerado é um baralho de 32 cartas, um problema do iComb pode ser:

Em um baralho de 32 cartas, quantas mãos de 4 cartas podem ser formadas com exatamente 1 Ás e 3 rainhas?”

Quadro 2: Sugestão de problema para o iComb

A fim de estabelecer uma linguagem formal para a estrutura de problemas, serão definidos alguns conceitos, utilizando o problema acima como exemplo.

Um *estágio* é definido por dois parâmetros, um conjunto de elementos e um número chamado *grandeza*: sendo D um conjunto de elementos, um *estágio* de D com *grandeza* g será representado por $[D, g]$, de modo que $[D, g] = \{d \subset D : |d| = g\}$. D será denominado de *domínio* de um estágio, sendo representado por $[D, g] = D$, enquanto a *grandeza* de um estágio será representada por $g([D, g]) = g$.

O problema do quadro 2 pode ser resolvido utilizando-se 2 estágios, D_1 formado pelas cartas que são áses e D_2 formado por 3 cartas de rainhas: $D_1 = \{\clubsuit^A, \heartsuit^A, \spadesuit^A, \diamondsuit^A\}$ e $D_2 = \{\clubsuit^Q, \heartsuit^Q, \spadesuit^Q, \diamondsuit^Q\}$. A partir do enunciado do problema, deduz-se que as grandezas dos estágios são respectivamente 1 e 3, ou seja, deve-se pegar elementos com cardinalidade 1 em D_1 e 3 em D_2 , resultando nos seguintes conjuntos:

$$\begin{aligned}
 [D_1, 1] &= \{\{\clubsuit^A\}, \{\heartsuit^A\}, \{\spadesuit^A\}, \{\diamondsuit^A\}\} \\
 [D_2, 3] &= \{\{\heartsuit^Q, \spadesuit^Q, \diamondsuit^Q\}, \{\clubsuit^Q, \spadesuit^Q, \diamondsuit^Q\}, \\
 &\quad \{\clubsuit^Q, \heartsuit^Q, \diamondsuit^Q\}, \{\clubsuit^Q, \heartsuit^Q, \spadesuit^Q\}\}.
 \end{aligned}$$

Assim, o conjunto solução pode ser obtido como o produto cartesiano $[D_1, 1] \otimes [D_2, 3]$ e resposta final ao problema (*quantos?*) é a cardinalidade deste conjunto produto (cujos elementos são formados por 4 cartas): $|[D_1, 1]| \times |[D_2, 3]| = 16$.

De modo geral, uma *solução* para um problema do iComb é gerada por um conjunto de estágios $\{S_1, S_2, \dots, S_k\}$, sendo que:

- R_1 . $S_1 \otimes S_2 \otimes \dots \otimes S_k$ é o conjunto de todas as configurações que atendem o enunciado.
- R_2 . S_1, S_2, \dots, S_k são dois a dois disjuntos.

A partir destes conceitos, pode-se provar uma equivalência entre soluções de problemas satisfazendo os dois itens acima, como aparece no teorema 1 [Eisenmann 2009].

Teorema 1 *Se $S = \{S_1, S_2, \dots, S_k\}$ é a solução para um problema do iComb, um conjunto de estágios $Z = \{Z_1, Z_2, \dots, Z_m\}$ também será uma solução se:*

1. $\{x \in S : 0 < g(x) < |\tilde{x}|\} = \{x \in Z : 0 < g(x) < |\tilde{x}|\}$
2. $\bigcup_{\substack{x \in S \\ g(x)=0}} \tilde{x} = \bigcup_{\substack{x \in Z \\ g(x)=0}} \tilde{x}$
3. $\bigcup_{\substack{x \in S \\ g(x)=|\tilde{x}|}} \tilde{x} = \bigcup_{\substack{x \in Z \\ g(x)=|\tilde{x}|}} \tilde{x}$

Assim, os problemas do iComb satisfazem às restrições R_1 e R_2 e a partir do teorema 1 foi possível implementar um avaliador automática para o iComb bastante eficiente (rápido).

5. Experimentos

O iComb foi avaliado do ponto de vista funcional [Nielsen e Mack 1994, Nielsen 2000] e didático com um grupo de 6 professores de matemática do ensino médio. O experimento consistiu na resolução de alguns exercícios, de forma integrada a um SGC, sem intervenção de instrutor.

Inicialmente foi mostrado o SGC, onde estavam os exercícios e então o instrutor abriu o iComb com o primeiro exercício, mostrando como resolvê-lo, como quebrar um problema em etapas e os botões necessário para completar a tarefa. Depois foi solicitado que os professores resolvessem os demais 9 exercícios¹⁰. Além do registro de soluções enviadas, foi observado como o grupo utilizava o iComb. Ao final foi distribuído um questionário para os professores avaliarem o entendimento do processo de resolução de

¹⁰Os exercícios apresentados foram: (1) Com um baralho de 32 cartas, quantas mãos de 2 cartas é possível formar com exatamente uma Dama e um Ás? (2) Com um baralho de 32 cartas, quantas mãos de 5 cartas é possível formar com exatamente 2 copas e 2 espadas? (3) Com um baralho de 52 cartas, quantas mãos de 8 cartas é possível formar com exatamente 1 ás, 2 reis e 2 valetes? (4) Com um baralho de 32 cartas, quantas mãos de 8 cartas é possível formar com exatamente 4 ases e 4 reis? (5) Com um baralho de 32 cartas, quantas mãos de 20 cartas é possível formar com 3 Ases, 4 cartas de espadas e 16 cartas vermelhas? (6) Com um baralho de 32 cartas, quantas mãos de 12 cartas é possível formar com o mínimo de sete cartas de copas, exatamente 4 ases, 2 cartas de espadas e nada mais? (7) Com um baralho de 32 cartas, quantas mãos de 18 cartas é possível formar com 6 cartas de ouros, no máximo 4 pretas, exatamente 4 valetes e exatamente 4 ases? (8) Com um baralho de 32 cartas, quantas mãos de 13 cartas é possível formar com no mínimo 1 carta de copas, exatamente 5 cartas pretas e exatamente 4 ases? (9) Com um baralho de 32 cartas, quantas mãos de 13 cartas é possível formar com 12 cartas vermelhas, exatamente 3 ases e no máximo 4 cartas de ouros? (10) Quantos times de 13 jogadores podemos formar com exatamente 5 jogadores do time azul e exatamente 5 jogadores do time verde?

um exercício com o iComb (o modelo de estágios e sua implementação na interface) e algumas questões qualitativas e quantitativas sobre o sistema. As questões apresentadas foram: (1) Você conseguiu resolver os exercícios propostos? 2) Quando o iComb apresentou mensagens de erro, facilitou na resolução no exercício? 3) Você acha que o iComb pode ajudar no ensino de combinatória? 4) Você tem alguma sugestão para melhorar o iComb? 5) Sobre a utilização do iComb, você acha que: a) fácil b) razoável c) difícil 6) De que modo você utilizaria o iComb em sala de aula? 7) Comentários Gerais.

No geral, a avaliação do iComb foi positiva com a percepção geral de que o sistema era uma boa idéia e útil para o ensino de combinatória. Cinco dos seis professores responderam que o iComb pode ajudar no ensino de combinatória. Houve um grande número de comentários com relação a quantidade reduzida de exercícios do iComb ilustrando o não entendimento do conceito de repositório de exercícios. De fato, o sistema iComb pode trabalhar com qualquer número de exercícios, não havendo qualquer referência aos exercícios no código fonte do programa. Este comentário dos professores reforça a importância da idéia de repositório de exercícios.

Críticas a qualidade das mensagens de erro foram feitas por boa parte dos professores. Além da deficiência do sistema, este conjunto de comentários mostra que a utilização do sistema iComb não é trivial e imediata, pois se fosse, os professores dariam menos importância as mensagens de erro obtidas. O fato de que 5 entre 6 professores avaliaram a utilização do iComb como de *razoável* dificuldade corrobora esta última observação.

6. Conclusões e Trabalhos Futuros

A implementação do iComb para proporcionar uma ferramenta para ensino e aprendizagem de combinatória na Web, foi baseada numa experiência bem sucedida do *Combien?*. Mesmo em sua primeira versão de teste, mostrou-se útil para um grupo de professores, que relataram que o utilizariam em aulas.

A proposta da arquitetura de *plugins* foi flexibilizar o sistema de maneira a permitir que novos universos e seus respectivos exercícios sejam criados. A motivação por trás desta idéia é a hipótese de que exercícios de determinados universos podem ser mais efetivos para a aprendizagem do que outros, dependendo do ambiente sócio cultural dos alunos em questão. De fato, alguns professores presentes no experimento realizado propuseram a criação de exercícios que não utilizassem baralhos ou cartas. A atual versão do iComb possui dois *plugins*, o universo “Baralho” e o universo “Futebol”, este último possivelmente mais ao gosto dos alunos brasileiros.

É importante destacar que uma das contribuições deste projeto foi a fundamentação matemática que resultou no teorema 1. Com este teorema foi possível implementar no iComb um mecanismo bastante rápido para avaliação automática, que consegue detectar erros antes mesmo do aluno finalizar sua solução. Deste modo o iComb consegue fornecer dicas ou advertências ao aluno, atendendo à necessidade de cursos na Web fornecerem resposta rápidas para os alunos.

A partir do experimento com especialistas no domínio (professores de matemática), estabelecemos como próxima melhoria no iComb, deixar mais claro em sua interface o significado dos estágios e como resolver um exercício neste modelo. Outro

ponto desejável é ampliar a categoria de problemas que podem ser examinados no iComb, tentando-se generalizar o teorema 1. Outro trabalho futuro a ser realizado com o iComb é construir uma rede de usuário-professores para produzir novos exercícios e testá-los com seus alunos.

O sistema está disponível para experimentação no endereço <http://www.matematica.br/icomb>. Está-se tentando resolver um problema de manter oculto o gabarito dos exercícios, de modo a ser possível também distribuir o fonte do iComb (como *software livre*).

Agradedimento

Leônidas O. Brandão foi parcialmente financiado pela FAPESP, projeto número 05/60647-1.

7. Referências

Referências

- BRASIL (1997). Parâmetros Curriculares Nacionais: Matemática. Ministério da Educação. <http://portal.mec.gov.br/seb/arquivos/pdf/livro03.pdf>.
- Bridge, P. e Appleyard, R. (2008). A comparison of electronic and paper-based assignment submission and feedback. *British Journal of Educational Technology*, (39(4)):644650.
- Eisenmann, A. (2009). Icomb: Um sistema para o ensino e aprendizagem de combinatória em ambiente web. Master's thesis, Universidade de São Paulo.
- Giroire, H., Le Calvez, F., e Tisseau, G. (2006). Benefits of Knowledge-Based Interactive Learning Environments: A Case in Combinatorics. *Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies*, pages 285–289.
- Laborder, J. M. e Bellemain, F. (1997). Cabri geometry ii. Technical report, Dallas: Texas Instruments.
- Le Calvez, F., Giroire, H., Duma, J., Tisseau, G., e Urtasun, M. (2003). Combien? a Software to Teach Students How to Solve Combinatorics Exercises. *Proceedings of Artificial Intelligence in Education, Sydney, Australia, IOS Press, Amsterdam*, pages 447–453.
- Morgado, A. C. O., Carvalho, J. B. P., Carvalho, P. C. P., e Fernandez, P. (1991). *Análise Combinatória e Probabilidade*. Coleção do Professor de Matemática. Sociedade Brasileira de Matemática, 9 edition.
- Nielsen, J. (2000). Why you only need to test with 5 users. *Jakob Nielsen's Alertbox*.
- Nielsen, J. e Mack, R. L. e. (1994). *Usability inspection methods*. NY: Wiley & Sons.
- Tisseau, G., Giroire, H., Le Calvez, F., Urtasun, M., e Duma, J. (2000). Design principles for a system to teach problem solving by modelling. *Lecture Notes in Computer Science*, 1839:393–402.
- VW (2009). Visual works. <http://www.cincomsmalltalk.com/userblogs/cincom/blogView>.