UNIVERSIDADE DE SÃO PAULO

SABNAG: Um sistema baseado em conhecimento para suporte aos usuários da biblioteca NAG

JÚLIA SAWAKI

MARIA CAROLINA MONARD

SOLANGE REZENDE RODRIGUES

Nº 10

NOTAS

Instituto de Ciências Matemáticas de São Carlos

Instituto de Ciências Matemáticas de São Carlos

ISSN - 0103-2577

SABNAG: Um sistema baseado em conhecimento para suporte aos usuários da biblioteca NAG

JÚLIA SAWAKI

MARIA CAROLINA MONARD

SOLANGE REZENDE RODRIGUES

Nº 10

N O T A S D O I C M S C

Série Computação

São Carlos

Jul. / 1994

SABNAG: Um Sistema Baseado em Conhecimento para Suporte aos Usuários da Biblioteca NAG

Júlia Sawaki[1]

Universidade Estadual Paulista
Instituto de Química
Departamento de Físico-Química
Araraquara - SP

Maria Carolina Monard[2]
Solange Rezende Rodrigues[2]

Universidade de São Paulo / ILTC
Instituto de Ciências Matemáticas de São Carlos
Departamento de Ciências de Computação e Estatística
São Carlos - SP

Resumo

A Biblioteca de Rotinas NAG é uma biblioteca bastante extensa e abrangente que fornece diversas rotinas para resolução de problemas de Análise Numérica, Otimização, Estatística, etc... Ainda que esta biblioteca esteja muito bem documentada, devido ao grande número de rotinas disponíveis, a tarefa de selecionar a rotina mais apropriada para resolução de um problema particular pode, às vezes, não ser simples para os usuários da NAG por exigir consulta a um grande volume de informações nos manuais.

Este trabalho apresenta as idéias utilizadas na implementação de um Sistema Baseado em Conhecimento -SBC- denominado SABNAG, que tem o objetivo de auxiliar os usuários da Biblioteca de rotinas Fortran NAG na procura da(s) melhor(es) rotina(s) para resolver um determinado problema. Além de encontrar a(s) rotina(s) apropriada(s), baseado nas informações fornecidas pelo usuário e no conhecimento que possui a respeito da Biblioteca NAG, o sistema explica *como* chegou a esta solução, *por que* necessita de uma determinada informação, bem como *o que acontece se* algumas informações são alteradas após ter encontrado uma solução.

O mecanismo de inferência e controle do sistema SABNAG foi projetado fazendo uso de um ambiente que auxilia na construção de núcleos específicos de SBCs. O sistema é auto-explicativo, sua Base de Conhecimento contém mais de 750 regras e foi implementado na linguagem de programação lógica Prolog para microcomputadores IBM-PC e compatíveis.

[1]Trabalho realizado com auxílio da CAPES e ILTC - Instituto de Lógica, Filosofia e Teoria da Ciência.

[2]Trabalho realizado com auxílio do CNPq e FAPESP - Processo Nº 92/2151-8

Conteúdo

1 Introdução **1**

2 Biblioteca de Rotinas Fortran NAG **2**

3 O Sistema SABNAG **3**

3.1 Aquisição de Conhecimento 3

3.2 Estruturação do Conhecimento 3

3.3 Organização da Base de Conhecimento 5

3.4 Ambiente Utilizado no Desenvolvimento do Sistema 6

3.5 Implementação do Sistema 8

4 Exemplos de Interação com o Sistema **9**

5 Conclusões **13**

1 Introdução

Os Sistemas Baseados em Conhecimento -SBC- ou, resumidamente, Sistemas de Conhecimento -SC- são programas de computador que resolvem, ou ajudam a resolver, problemas que geralmente requerem o uso de inteligência humana na sua resolução. Os Sistemas de Conhecimento diferem dos sistemas convencionais na forma em que são organizados, na maneira como incorporam os conhecimentos, na forma que manipulam esse conhecimento e na impressão que eles causam nos usuários com quem interagem. Além disso, devem ser capazes de trabalhar com conhecimento impreciso.

Os Sistemas Especialistas -SE- constituem uma classe muito especial dos Sistemas Baseados em Conhecimento. Há um crescente consenso na comunidade de Inteligência Artificial sobre as características que um Sistema Baseado em Conhecimento deve ter para ser chamado de Sistema Especialista [3]. A mais importante delas é, sem dúvida, a capacidade de fornecer uma solução a nível de um especialista para problemas complexos. A segunda característica é que o procedimento de resolução do problema deve ser compreensível pelo usuário do sistema. A terceira característica é a flexibilidade, isto é, o sistema deve ser flexível o suficiente para poder aplicar seu conhecimento a novos problemas que surjam no seu domínio. Além disso, os SEs devem ser capazes de explicar a sua conduta, ou seja, por que necessitam de novas informações externas, como eles chegaram a certa conclusão, etc... [5]

Frequentemente, o termo Sistema Baseado em Conhecimento é usado como alternativa para o termo Sistema Especialista, que foi muito pretensioso e ganhou certa conotação negativa. Provavelmente, o termo SBC é mais apropriado, porque refere-se a sistemas menos restritos, além de ter conotação menos pretensiosa em relação à verdadeira capacidade destes sistemas[3].

Neste trabalho é apresentado um Sistema Baseado em Conhecimento, por nós implementado, denominado SABNAG, que auxilia o usuário da Biblioteca de rotinas Fortran NAG na busca da(s) melhor(es) rotina(s) para resolver o seu problema.

O sistema SABNAG é auto-explicativo, interroga o usuário e, com base nas respostas e no "conhecimento" que possui referente a NAG, apresenta suas conclusões. Este sistema admite respostas do tipo múltipla escolha, afirmativas, negativas, *não-sei* e *por-que*. No caso do usuário responder *não-sei*, isto é, ele não tem a informação que lhe é requisitada, o sistema possui conhecimento para ainda neste caso, continuar procurando por alguma solução factível. No caso do usuário responder *por-que*, o sistema explica os motivos da pergunta. Finalmente, após aconselhar a(s) rotina(s) mais apropriada(s), o sistema SABNAG fornece ao usuário a opção de saber *como* chegou a essa conclusão e *o que acontece se* o usuário trocar algumas das suas respostas.

O trabalho está organizado da seguinte forma: na seção 2 é descrita suscintamente a organização da Biblioteca NAG. Na seção 3 é apresentada a aquisição de conhecimento, a estruturação de conhecimento, a organização da Base de Conhecimento NAG, o am-

[3]No que se segue, como na maioria dos trabalhos nesta área, os termos Sistemas Baseados em Conhecimento e Sistemas Especialistas serão usados indistintamente.

biente utilizado na implementação do sistema SABNAG, bem como a implementação deste sistema. Na seção 4 são apresentados alguns exemplos de interação com o sistema e, na seção 5 são apresentadas as conclusões deste trabalho.

2 Biblioteca de Rotinas Fortran NAG

A Biblioteca de Rotinas NAG é uma coleção de rotinas para a solução de problemas numéricos e estatísticos por computador. Periodicamente uma nova versão da Biblioteca NAG é lançada e, então, novos algoritmos são adicionados, correções ou melhoramentos são feitos nas rotinas já existentes e algumas rotinas são substituídas por outras mais aperfeiçoadas. A Biblioteca NAG vem acompanhada de uma vasta documentação. O principal documento é o Manual da Biblioteca que auxilia o usuário na identificação do problema, seleção de algoritmos e uso das rotinas.

No desenvolvimento deste trabalho, foi usada a versão 14 do Manual da Biblioteca de Rotinas Fortran NAG [8], disponível no CISC[4] da USP, campus de São Carlos. Este Manual consta de 8 volumes de aproximadamente 300 páginas cada. As 852 rotinas disponíveis nesta versão estão divididas em 37 capítulos — Tabela 1 — de acordo com o assunto tratado.

Volume	Capítulos	Nº de Rotinas
1	A02 - D01	76
2	D02	66
3	D03 - E04J	62
4	E04K - F01	99
5	F02 - F06	241
6	G01 - G02	91
7	G03 - G13	99
8	H - X05	118
Total	37	852

Tabela 1: Estrutura do Manual da Biblioteca de Rotinas Fortran NAG - versão 14

Cada capítulo do Manual da Biblioteca NAG tem uma introdução e uma recomendação para a escolha das rotinas a ele pertencentes. Para cada rotina são fornecidas informações específicas da rotina tais como: propósito, especificação, descrição, parâmetros de entrada e saída.

O nome do capítulo é constituído por um único caracter, ou um caracter seguido por 2 dígitos, e cada capítulo possui também um título. Os nomes das rotinas são constituídos por 6 caracteres, começando com o(s) caracter(es) do nome do capítulo a que pertence, e terminando com o caracter F (de Fortran).

A seleção manual de uma rotina apropriada para executar uma função específica é uma

[4]CISC - Centro de Informática de São Carlos

tarefa demorada. Com o objetivo de auxiliar o usuário na busca da rotina apropriada, alguns capítulos do Manual da Biblioteca NAG fornecem árvores de decisão, que foram utilizadas na implementação das regras que constituem a Base de Conhecimento do sistema SABNAG.

3 O Sistema SABNAG

3.1 Aquisição de Conhecimento

Em Sistemas Baseados em Conhecimento o processo de aquisição de conhecimento — extração do conhecimento do especialista ou livros — constitui o gargalo no seu desenvolvimento [2]. Devido a este fato, existem várias metodologias e ferramentas para auxiliar esta difícil tarefa. Porém devido ao fato de que no nosso ambiente de pesquisa não dispomos de nenhuma destas ferramentas, a aquisição de conhecimento para o sistema SABNAG foi realizada sem o auxílio deste tipo de ferramentas.

A Base de Conhecimento NAG foi construída extraindo-se o conhecimento contido no Manual da Biblioteca de Rotinas Fortran NAG - versão 14. O fato da fonte de conhecimento não ser um especialista humano mas sim um documento, trouxe certas vantagens e algumas desvantagens no processo de aquisição de conhecimento.

Vantagens porque o Manual da Biblioteca NAG está disponível para ser consultado a qualquer hora, o que muitas vezes não acontece quando a fonte de conhecimento é um especialista humano. Além disso, no Manual o conhecimento já está organizado de alguma forma, o que facilita a tarefa de aquisição de conhecimento realizada pelo Engenheiro de Conhecimento. Isto porque quando a fonte de conhecimento é um especialista humano, na maioria das vezes o conhecimento que este possui do assunto não está organizado, dificultando consideravelmente a aquisição deste conhecimento.

A desvantagem de extrair o conhecimento de um documento é que, às vezes, o Engenheiro de Conhecimento sente a necessidade de obter mais informações além daquelas contidas no documento. Neste caso, foi solicitado o auxílio de um especialista da área.

A Base de Conhecimento do sistema SABNAG é bastante abrangente, incluindo o conhecimento necessário para identificar qualquer rotina de Análise Numérica e Otimização, cerca de 70% de todo o conhecimento contido na Biblioteca NAG. A escolha pelas rotinas de Análise Numérica e Otimização foi devido ao fato de serem as rotinas da Bibliteca NAG mais procuradas pelos usuários do campus de São Carlos.

3.2 Estruturação do Conhecimento

Tendo em vista a construção de uma Base de Conhecimento estruturada para a Biblioteca NAG, foi realizado um estudo cuidadoso do cohecimento contido na Biblioteca. Assim, foi possível estruturar o conhecimento em 11 classes, cada uma delas correspon-

dendo a uma grande área de conhecimento tal como: Aritmética Complexa, Integração e Diferenciação Numérica, Álgebra Linear Numérica, Estatística, Pesquisa Operacional, etc. Além disso, foi verificado que cada uma destas classes pode ser dividida em subclasses que constituem 37 subáreas, correspondentes aos 37 capítulos do Manual da NAG [11]. A Figura 1 mostra como a Biblioteca de Rotinas Fortran NAG foi por nós estruturada.

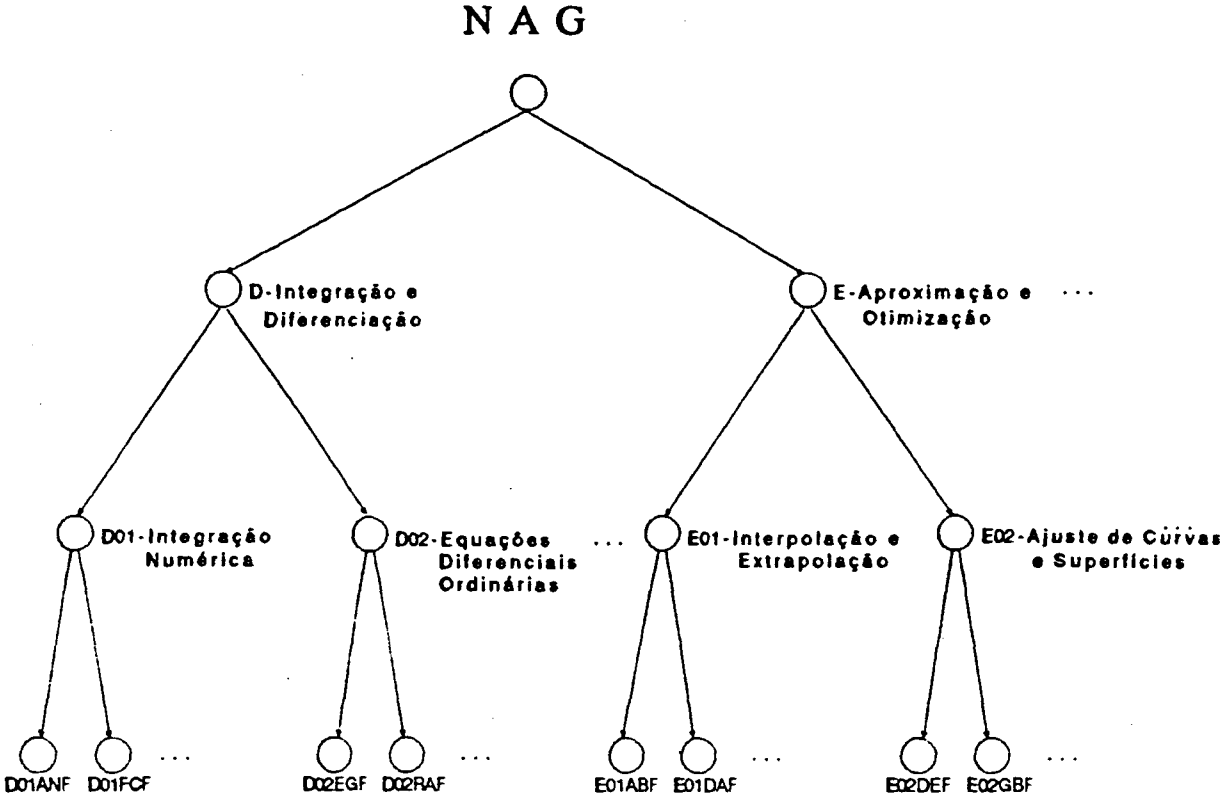

Figura 1: Estruturação do Conhecimento da Biblioteca NAG em Áreas e Subáreas

Esta estrutura da Biblioteca NAG auxilia no processo de busca de uma rotina para uma função específica, pois quando o usuário informa a sua área de interesse — que é uma das primeiras informações requerida pelo sistema — ele está restringindo, automaticamente, o espaço de busca na seleção da rotina apropriada. Este subespaço de busca é novamente subdividido quando o usuário informa ao sistema a subárea de interesse.

O exemplo a seguir ilustra a classificação de seis subáreas — F01, F02, F03, F04, F05 e F06 — da área F - Álgebra Linear Numérica, segundo a estruturação citada acima.

- **Área:**

  F - Álgebra Linear Numérica

- **Subáreas:**

  F01 - Operação e Inversão de Matrizes (64 rotinas)

  F02 - Auto-valores e Auto-vetores (40 rotinas)

  F03 - Determinantes (9 rotinas)

  F04 - Equações Lineares Simultâneas (37 rotinas)

  F05 - Ortogonalização (1 rotina)

  F06 - Rotinas de Suporte à Álgebra Linear (154 rotinas)

Por exemplo, se a subárea de interesse for Determinantes, a busca é realizada somente sobre as 9 rotinas desta subárea.

3.3 Organização da Base de Conhecimento

A Base de Conhecimento implementada é bastante extensa e contém o conhecimento necessário para o sistema identificar mais de 500 rotinas da NAG. Na construção da BC foram levados em consideração os seguintes aspectos:

- o problema de limitação de memória do equipamento em uso;
- o problema de eventuais extensões da Base de Conhecimento devido à publicação de novas versões da Biblioteca NAG;
- o problema de restringir tanto quanto possível o espaço de busca do Motor de Inferência, e assim minimizar o tempo de resposta do sistema.

A fim de solucionar estes problemas, o conhecimento da Biblioteca NAG foi dividido em sub-bases hierarquicas de conhecimento completamente independentes, como é mostrado na Figura 2.

O conhecimento necessário para identificar a área e a subárea de interesse foi reunido em uma sub-base no topo da hierarquia. Essa sub-base geral contém o conhecimento necessário para acessar o conhecimento específico de cada subárea nas sub-bases correspondentes.

Assim, foram solucionados os problemas antes mencionados, uma vez que:

- apesar da Base de Conhecimento ser bastante extensa, o problema de limitação de memória do equipamento foi contornado, pois o conhecimento que precisa estar na memória do computador é apenas o conhecimento geral, necessário para realizar a busca e determinar a subárea de interesse, e o conhecimento específico da subárea selecionada.

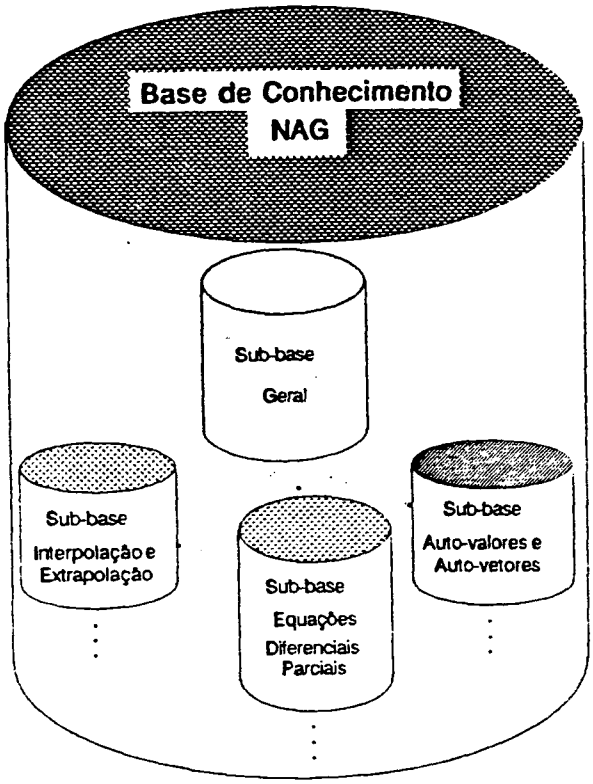

Figura 2: Organização da Base de Conhecimento NAG

- eventuais acréscimos na Biblioteca NAG implicam em extensão da BC. Caso seja acrescentada uma nova subárea na Biblioteca NAG, deve ser criada uma nova sub-base de conhecimento específica desta subárea, sem necessidade de alteração das demais sub-bases. Caso seja acrescentada uma nova rotina, ou seja feita alguma alteração na subárea já existente, somente a sub-base referente a esta subárea deve ser alterada.

- o espaço de busca foi restringido bastante, pois a busca é realizada somente sobre o conhecimento armazenado dentro da sub-base correspondente.

3.4 Ambiente Utilizado no Desenvolvimento do Sistema

A construção do Núcleo de um SE pode ser facilitada se for realizada em um ambiente que permite articular, bem como alterar, os diversos subsistemas que o constituem, tal que eles possuam características apropriadas para manipular Bases de Conhecimentos diferentes.

Na construção do sistema SABNAG foi utilizado um ambiente desenvolvido no ICMSC-USP-São Carlos [10]. Este ambiente está implementado na linguagem de programação lógica Prolog, faz extenso uso de meta-interpretação e é dirigido a usuários não leigos em Sistemas Baseados em Conhecimento e Prolog. O ambiente — ilustrado na Figura 3 — está apoiado na estrutura básica de Sistemas Especialistas que é constituída por:

- Base de Conhecimento -BC- que contém os fatos relativos à área de especialização, as regras que descrevem as relações no domínio de conhecimento, e os métodos heurísticos para a resolução de problemas no domínio específico.
- Base de Dados -BD- que constitui a memória de trabalho do sistema, contém os dados de entrada do problema, as respostas fornecidas pelo usuário, as asserções que foram deduzidas da BC, etc.
- O Núcleo do Sistema Especialista -NSE- que é constituído por três módulos descritos a seguir, é responsável pela interação com o meio externo ao Sistema Especialista, pelo processamento do conhecimento usando uma linha de raciocínio e, também, pela justificação e explicação das novas conclusões obtidas através dessa linha de raciocínio do sistema.

  Os módulos que constituem o NSE são:

  1. Módulo Coletor de Dados -MCD- responsável pela comunicação com o usuário e/ou outros sistemas [9];
  2. Motor de Inferência -MI- que possui uma linha de raciocínio e processa o conteúdo da BC e da BD. O MI do ambiente utiliza raciocínio "backward chaining" para provar as metas [6] ;
  3. Módulo de Explicação -ME- responsável pela explicação de *como* o MI chega a certas conclusões, *por que* faz certas perguntas, etc.

No Nível superior do ambiente tem-se a implementação completa do Núcleo de um SE, que inclui todas as opções nele implementadas. Porém, se a Base de Conhecimento a ser processada não fizer uso de todas as opções fornecidas pelo NSE do Nível superior, este ambiente permite ao projetista do sistema construir um Núcleo específico[5] que utilize apenas as opções necessárias para esta aplicação, retirando os meta-interpretadores não relevantes [4]. Este Núcleo restrito — que corresponde a um subconjunto do Núcleo ilustrado no Nível superior — é o mostrado no Nível 1 da Figura 3.

Este ambiente também permite, caso seja necessário, alterar o NSE para adequá-lo ao problema específico — Nível 2 da Figura 3 [7]. Tais alterações incluem: alteração na forma de gravar e acessar a Base de Dados, alteração na estrutura da informação manipulada pelo Módulo de Explicação, alteração no modelo usado pelo Motor de Inferência para processar raciocínio incerto, etc.

[5] A vantagem do uso de um Núcleo específico restrito ao problema é o aprimoramento no tempo de resposta do sistema final.

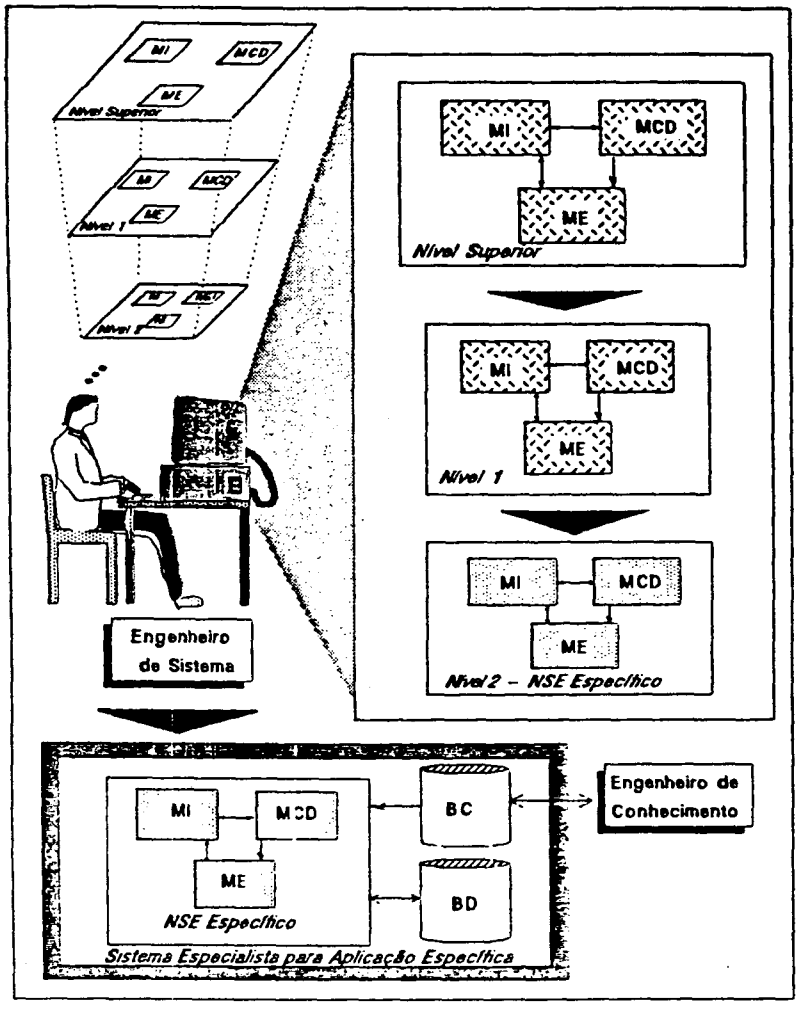

Figura 3: Ambiente para Auxiliar a Construção de Núcleos Específicos [7]

3.5 Implementação do Sistema

Na construção do sistema SABNAG, utilizando o ambiente acima descrito, foi construído um NSE específico — Nível 2 da Figura 3 — para manipular a Base de Conhecimento NAG por nós implementada.

Na fase de aquisição de conhecimento e construção das regras de conhecimento[6] foi tomado muito cuidado para construir regras que traduzam o conhecimento da melhor

[6]O conhecimento está representado por regras do tipo Se < Condições > então < Conclusões >.

forma possível. Nesta fase, as árvores de decisão fornecidas por alguns capítulos do Manual da Biblioteca NAG foram muito úteis. Além do conhecimento contido no Manual da NAG, foi também solicitado auxílio do conhecimento de pesquisadores da área.

Devido a estruturação do conhecimento adotado — seção 3.2 — foi possível implementar as regras de conhecimento de cada uma das subáreas em arquivos diferentes, que correspondem às sub-bases citadas na seção 3.3. Além desses arquivos contendo o conhecimento específico de cada subárea, foi implementado o módulo principal que corresponde à sub-base geral da Figura 2. Este módulo é responsável por ativar o Motor de Inferência do sistema, gerenciar os arquivos da Base de Conhecimento e também é responsável pela interface com o usuário.

Utilizando janelas e caixas de diálogo, o sistema inicialmente interroga a área e em seguida a subárea de interesse do usuário. Uma vez determinada a subárea de interesse, o sistema carrega na memória o conhecimento específico da subárea selecionada.

Na implementação do conhecimento específico da subárea, foi criada inicialmente uma tabela condensada de regras de conhecimento e uma lista contendo as perguntas que são feitas ao usuário. Esta tabela de regras e a lista de perguntas constituíram sistematicamente a primeira forma de representação do conhecimento contido no Manual da Biblioteca NAG. A partir desta tabela foi realizada a implementação das regras.

Após verificada a validade das regras, a tabela condensada de regras e a lista de perguntas foram utilizadas como documentação da BC NAG, uma vez que juntas fornecem uma visualização e documentação conveniente das regras de conhecimento.

Deve ser ressaltado que nas subáreas onde o número de rotinas disponíveis é razoavelmente grande, as rotinas foram ainda subdivididas em classes de acordo com o assunto de interesse do usuário dentro daquela subárea.

A Tabela 2 ilustra todas as áreas e as respectivas subáreas implementadas neste trabalho, que constituem aproximadamente 70% do conhecimento contido no Manual da Biblioteca de Rotinas Fortran NAG.

Finalmente, diversos testes foram realizados para verificar se as regras implementadas simulavam satisfatoriamente o conhecimento contido no Manual da Biblioteca NAG.

4 Exemplos de Interação com o Sistema

Nesta seção são apresentados alguns exemplos de interação com o sistema SABNAG.

A Figura 4 ilustra o caso em que o usuário seleciona a área "Integração e Diferenciação"; dentro desta área a subárea "Integração Numérica" e o assunto "Integral sobre um Intervalo Finito". Após selecionado o assunto de interesse do usuário, o sistema faz sucessivas perguntas. Todas as perguntas admitem as respostas *nao-sei* e *por-que*. No exemplo da Figura 4, *nao-sei* é a resposta para uma pergunta e *por-que* para outra.

CA	Nome da Área	CS	Nome da Subárea
A	Aritmética Complexa	A02	Aritmética Complexa
C	Raízes de Equações e Soma de Séries	C02	Raízes de Equações Polinomiais
		C05	Raízes de Uma ou Mais Equações Transcendentais
		C06	Soma de Séries
D	Integração e Diferenciação	D01	Integração Numérica
		D02	Equações Diferenciais Ordinárias
		D03	Equações Diferenciais Parciais
		D04	Diferenciação Numérica
		D05	Equações Integrais
E	Aproximação e Otimização	E01	Interpolação e Extrapolação
		E02	Ajuste de Curvas e Superfícies
		E04	Minimização e Maximização de Funções
F	Álgebra Linear Numérica	F01	Operação e Inversão de Matrizes
		F02	Auto-valores e Auto-vetores
		F03	Determinantes
		F04	Equações Lineares Simultâneas
		F05	Ortogonalização
		F06	Rotinas de Suporte à Álgebra Linear
H	Pesquisa Operacional	H	Pesquisa Operacional
M	Ordenação	M01	Ordenação

CA = Código da Área **CS = Código da Subárea**

Tabela 2: Áreas e Subáreas da Biblioteca NAG codificadas

Neste último caso, o sistema explica o motivo pelo qual ele está fazendo esta pergunta mostrando a regra que está tentando provar.

Deve ser ressaltado que o usuário pode voltar a responder *por-que* para a mesma pergunta. Com isso, o sistema justifica mostrando a regra imediatamente acima no espaço de busca. Este processo pode ser repetido até que o sistema responda que não possui mais informação para fornecer. Isso significa que o sistema já mostrou todo o caminho percorrido até o momento, na tentativa de provar a meta original — aconselhar uma rotina ideal para o problema do usuário.

Quando o sistema chega à solução do problema, mostra a solução e dá a opção de saber como chegou a esta solução ou trocar algumas das informações fornecidas pelo usuário. Caso o usuário selecione o primeiro ítem — Figura 5 —, o sistema explica ao usuário *como* conseguiu chegar a esta solução. Para isso, mostra a regra (que contém a meta principal) que ele provou, e pergunta se o usuário deseja mais explicações. Caso seja de interesse do usuário, o sistema explica também como esta regra foi provada, e assim sucessivamente.

```
Area : Integracao e Diferenciacao
Subarea : Integracao Numerica

Voce dispoe da forma funcional do integrando (s/n) ? s

Voce deseja integracao indefinida (s/n) ? nao-sei

Voce esta interessado na eficiencia de integrais simples (s/n) ? por-que

SE
   dispoe da forma funcional do integrando    E
   nao deseja integracao indefinida    ou
   nao sabe se deseja integracao indefinida    E
   nao estah interessado na eficiencia de integrais simples
ENTAO
   a rotina recomendada eh DQ1NAF ou DU1NTF ou DU1NNF

Voce esta interessado na eficiencia de integrais simples (s/n) ? s
```

Figura 4: Explicação de *por-que* o Sistema fez Certa Pergunta

```
Area : Integracao e Diferenciacao
Subarea : Integracao Numerica

Aconselho a rotina DQ1NAF ou DQ1DAF

****** EXPLICACAO DE COMO O SISTEMA CHEGOU A SOLUCAO ******

Quer que explique os consideranos dessas condicoes (s/n) ? s

Voce disse que estah interessado em integral sobre um intervalo finito  E
Considerando que
   dispoe da forma funcional do integrando  E
   nao sabe se deseja integracao indefinida  E
   nao sabe se estah interessado na eficiencia de integrais simples  E
   o integrando eh bem comportado sem levar em conta a funcao peso  E
   nao deseja conhecer os pesos e as abscissas usados para a integracao
Conclui-se que a rotina recomendada eh DQ1NAF ou DQ1DAF

Quer que explique os consideranos dessas condicoes (s/n) ?
```

Figura 5: Explicação de *como* o Sistema chegou a uma Conclusão

Caso o usuário selecione o segundo ítem, *what if* — Figura 6 —, o sistema mostra todas as informações fornecidas pelo usuário, e pede para marcar a(s) informação(ões) que ele deseja alterar. Marcada a(s) informação(ões), o sistema apaga essas informações e tenta provar novamente a meta, ou seja, começa uma nova interação com o usuário — Figura 7 — lembrando as informações que permaneceram inalteradas.

```
Area : Integracao e Diferenciacao
Subarea : Integracao Numerica

Aconselho a rotina DQINAF ou DQIDAF

Marque com a tecla de <espaco>, as informacoes que voce deseja alterar

J estah interessado em Integral sobre um Intervalo Finito
  dispoe da forma funcional do integrando
  nao sabe se deseja integracao indefinida
  nao sabe se estah interessado na eficiencia de integrais simples
J o integrando eh ben comportado sem levar em conta a funcao peso
J nao deseja conhecer os pesos e as abscissas usadas para a integracao

Tecle <ENTER> apos marcar os itens
```

Figura 6: Alterando Informações

```
Area : Integracao e Diferenciacao
Subarea : Integracao Numerica

O integrando eh ben comportado, isto eh, pode ser aproximado por um polinomio
de grau moderado, a menos da funcao peso (s/n) ? n

O integrando eh razoavelmente suave (s/n) ? s

Voce requer precisao muito grande (s/n) ? s

O integrando tem singularidades, picos acentuados ou descontinuidades nos pon
tos conhecidos que nao sao limites finitos de integracao (s/n) ? s
```

Figura 7: Tentando Nova Prova

Deve ser observado que o sistema fará somente as perguntas necessárias para provar a meta. Isso quer dizer que as perguntas realizadas podem não incluir todas as informações que o usuário marcou para alterar, pois no novo contexto essas informações podem ser irrelevantes. Assim, baseado nas informações anteriores não alteradas e nas novas informações, o sistema aconselha a(s) rotina(s) ao usuário.

Isto pode ser visto claramente no exemplo mostrado na Figura 6 em que são marcadas duas informações para alterar. Porém, como mostrado na Figura 7, a segunda

informação não é perguntada novamente, já que não é utilizada no novo contexto.

5 Conclusões

O sistema SABNAG é um sistema auto-explicativo de fácil uso, que substitui a tarefa de consulta a um grande volume de informações no Manual da Biblioteca NAG.

A construção da Base de Conhecimento NAG foi bastante trabalhosa devido, em parte, ao fato de ter sido realizada manualmente sem o auxílio de nenhuma ferramenta para este tipo de tarefa. O código fonte completo do sistema SABNAG — implementado em Arity-Prolog [1] — contendo toda documentação, ocupa aproximadamente 600 Kbytes. A Base de Conhecimento contém mais de 750 regras e a definição de aproximadamente 250 perguntas distintas para obtenção de informações do usuário.

É importante salientar que apesar da Base de Conhecimento ser bastante extensa, é viável a execução deste sistema em equipamentos que possuem pouca memória, devido à forma como o conhecimento foi organizado em sub-bases diferentes e disjuntas.

Agradecimentos: à Dra. Célia Maria Finazzi de Andrade pelo seu auxílio na aquisição do conhecimento.

Referências

[1] Arity Corporation, *The Arity/Prolog Programming Language.* 1990.

[2] Aussenac, N.; Boy, G., *Lecture Notes in Artificial Intelligence, 723.* Knowledge Acquisition for Knowledge-Based Systems. Berlin, Springer, 1993.

[3] Bielawski, L.; Lewand, R., *Intelligent Systems Design: Integrating Expert Systems, Hypermedia, and Database Technologies.* John Wiley, 1991.

[4] Franco, J.L., *Sobre Avaliação Parcial de Meta-Interpretadores e Geração de Núcleos Específicos de Sistemas Especialistas.* Dissertação de Mestrado, ICMSC-USP, 1991.

[5] Luger, G.F; Stubllefield, W.A., *Artificial Intelligence and the Design of Expert Systems.* Benjamin, 1989.

[6] Monard, M.C.; Rodrigues, S.R., *Implementação Lógica de um Motor de Inferência com Raciocínio Backward Chaining para a Construção de Sistemas Especialistas.* Notas do ICMSC-USP, Nº 51, 1989.

[7] Monard, M.C.; Rodrigues, S.R., *An Environment for Developing Expert System's Shells.* Expert Systems World Congress Proceedings, Liebowitz J. (Ed.), Pergamon Press, Vol. 3, pp 1757-1765, 1991.

[8] *NAG Fortran Library Manual - Mark 14*. The Numerical Algorithms Group Limited, April 1990.

[9] Rodrigues, S.R; Monard, M.C., *Um Subsistema de um Núcleo de Sistema Especialista para Interrogar o Usuário Admitindo Diversas Categorias de Perguntas.* Anais VI SBIA, pp 78-93, 1989.

[10] Rodrigues, S.R., *Um Ambiente para Auxiliar a Construção de Núcleos de Sistemas Especialistas.* Dissertação de Mestrado, ICMSC-USP, 1990.

[11] Sawaki, J., *Um Sistema Baseado em Conhecimento para Apoio aos Usuários da Bibliteca NAG.* Dissertação de Mestrado, ICMSC-USP, 1991.

NOTAS DO ICMSC

Serie computaçao

Nº 009/94 NICOLETTI, M.C.; MONARD, M.C. Learning restricted Horn clauses: some considerations on the ij-determination concept

Nº 008/94 TOME, M.F.; DUFFY, B. GENSMAC: a numerical method for solving unsteady non-newtonian free surface flows

Nº 007/94 TOME, M.F.; MCKEE, S. Numerical simulation of viscous fluid: buckling of planar jets

Nº 006/94 MASIERO, P.C.; OLIVEIRA, M.C.F.; GERMANO, F.S.R; PIERRI,G.
Authoring and searching in dynamically growing hypertext data bases

Nº 005/94 NICOLETTI, M.C.; MONARD, M.C. Limiting the background knowledge in inductive logic programming

Nº 004/93 NICOLETTI, M.C.; MONARD, M.C. Learning horn clauses using the ILP system GOLEM

Nº 003/93 ARENALES, M.N.; MORABITO, R.N. An and/or graph approach to the solution of two-dimensional non-guillotine cutting problems

Nº 002/93 NICOLETTI, M.C.; MONARD, M.C. Herbrand interpretation model and least model within the framework of logic programming

Nº 001/93 MORABITO, R.; ARENALES, M.N. An and/or graph approach to the container loading problem