Nos testes foram utilizados dois estilos diferentes, ligados a linguagens simbólicas com diferentes apresentações.

Pelos resultados, menos de 50% dos alunos dominam a linguagem ligada ao estilo algébrico, menos de 40% dominam a linguagem que representa o estilo' geométrico e menos de 25% associam a linguagem do estilo algébrico à linguagem do estilo geométrico (interpretam e transferem).

Relacionando a fundamentação teórica dessa ' pesquisa com os resultados obtidos, concluimos:

1) A maioria dos alunos da experiência que ' deveria estar, pela sua idade cronológica e nível de escolaridade, no estágio das ' operações formais ou do pensamento adulto, ainda não a alcançou. Podemos afirmar que esses alunos, pelos tipos de erros cometidos, se encontraram no estágio das operações concretas, segundo Piaget.

2) A maioria dos alunos não interpreta o simbolismo como linguagem significativa e, ' portanto, não tem o domínio da linguagem' matemática, isto é, desconhece o discurso matemático.

   Para melhor aprendizagem, citaremos as opiniões de Piaget, Adler e Dieudonné: Segundo Piaget, existem três fatôres que desempenham um papel na preparação do individuo para avançar de um estágio mental para o seguinte:

1- A maturação do sistema nervoso.
2- A experiência adquirida na inteiração com o meio físico.
3- Influência do meio social.

Para Adler, dois critérios devem ser usados na seleção das experiências matemáticas que serão proporcionadas ao aluno, em qualquer idade:

a) devem ser experiências nas quais o aluno tenha possibilidade de trabalhar em função do estágio de desenvolvimento mental que atingiu.
b) essas experiências devem também preparar o aluno para atingir o estágio seguinte.

Segundo Dieudonné, não devemos camuflar o caráter abstrato da Matemática. O importante é colocar essa abstração em sintonia com o desenvolvimento mental dos alunos, tornando-a mais atraente possível.
EXEMPLO: apresentar para crianças de 10 anos, problemas de aritmética, cujas soluções são efetuadas por meio de equações lineares a uma incógnita, camuflando as variáveis, pela utilização de símbolos conhecidos (quadrado, triângulo, etc.). Numa idade em que o nível mental dos alunos não permite assimilar tal conceito, isto leva à automatização. Consequentemente, a criança terá mais dificuldade de adquirir esse

conceito no momento oportuno, pois já foi automatizado.

Sem dúvida, o processo de ensino que consiste na apresentação de receitas que, ainda hoje, são encontradas da mesma maneira, conduz ao adestramento. Contra isso, Dieudonné lança o seu protesto.

A partir dos nossos resultados e em vista das opiniões citadas, podemos propor:

1) Ao introduzir um novo conceito para os alunos devemos verificar se a maioria deles possui todos os pré-requisitos para a aquisição do mesmo.

2) O aluno deve ser <u>o autor da prática</u>, isto é, deve aprender a criar o seu próprio estilo e representá-lo simbolicamente em cada aquisição de conteúdo.

Essa prática deve ter o significado de uma redescoberta e não ser um processo automatizado. O aluno deve, portanto aprender a utilizar e interpretar o simbolismo como linguagem significativa dominando a linguagem matemática.

3) A formulação das questões em Matemática deve ser feita de maneira específica não permitindo ambiguidades.

4) As descobertas de diferentes caminhos para as soluções de situações - problema levam à ativação das estruturas mentais.

5) Os erros típicos de pensamento, característicos do estágio de desenvolvimento dos alu

nos, devem ser entendidos pelos professores.

6) O professor deve procurar minimizar o hiato existente entre a percepção e a formação de uma ação mental. É importante ampliar, promover e enriquecer a ação mental que se está desenvolvendo, pelo uso frequente de dados perceptíveis e de ações concretas.

7) Os professores de Matemática precisam dominar o discurso da linguagem corrente e da linguagem matemática.
   O uso da linguagem corrente não deve distorcer o pensamento e a linguagem matemática.

BIBLIOGRAFIA

(1) ADLER I.
MATEMÁTICA E DESENVOLVIMENTO MENTAL
EDITORA CULTRIX
SÃO PAULO - 1970

(2) BARTHES R.
ELEMENTOS DE SEMIOLOGIA
EDITORA CULTRIX
SÃO PAULO - 1971

(3) DUVAL R. E PLUVINAGE F.
DÉMARCHES INDIVIDUELLES DE RÉPONSE EN MATHÉMATIQUE
ARTIGO

(4) FISCHER J.P.
LA PERCEPTION DES PROBLÈMES SOUSTRACTIFS AUX DÉBUTS DE L'APPRENTISSAGE DE LA SOUSTRACTION
TESE APRESENTADA NA UNIVERSIDADE DE NANCY (FRANÇA) PARA OBTER O GRAU DE DOUTOR NO 3º CICLO.
1979

(5) FILLOY E., OCA V.M., RIESTRA J. E SENDEROS G.
MEDICION Y SISTEMAS DE NUMERACTION
TRABALHO REALIZADO NO CAMPO DE EDUCAÇÃO MATEMÁTICA NA FRANÇA E NO MÉXICO.
1977 E 1978

(6) GLAESER G.

LA DIDACTÎQUE EXPÉRIMENTALE DES MATHÉMATIQUES

COURS DE 3º CYCLE - UNIVERSITÉ LOUIS PASTEUR

STRASBOURG - 1979 - 1980

(7) GRANGER G.G.

FILOSOFIA DO ESTILO

EDITORA PERSPECTIVA E DITORA DA UNIVERSIDADE DE SÃO PAULO

SÃO PAULO - 1974

(8) HITT F.

COMPORTEMENT DE "RETOUR EN ARRIERE" APRÈS LA DECOUVERTE D'UNE CONTRADICTION

TESE APRESENTADA NA UNIVERSIDADE LOUIS PASTEUR DE STRASBOURG (FRANÇA) PARA OBTER O GRAU DE DOUTOR NO 3º CICLO .

STRASBOURG - 1978

(9) KARPLUS R.

PROCEEDINGS OF THE FOURTH INTERNATIONAL CONFERENCE FOR THE PSYCHOLOGY OF MATHEMATICS EDUCATION

UNIVERSITY OF CALIFORNIA

BERKELEY - CALIFORNIA - 1980

(10) MARTINET A.

LA LINGUISTÎQUE SYNCHONÎQUE

PRESSES UNIVERSITAIRES DE FRANCE

PARIS - 1968

(11) MARTINET A.
LINGUISTIQUE - GUIDE ALPHABETÎQUE
EDITTIONS DENOËL
PARIS - 1969

(12) NICK E. E KELLNER S.R.DE O.
FUNDAMENTOS DE ESTATÌSTICA PARA AS CIÊNCIAS DO COMPORTA
MENTO
EDITORA RENES
RIO DE JANEIRO - 1971

(13) NICK E. E RODRIGUES H.
MODELOS EM PSICOLOGIA
ZAHAR EDITORES
RIO DE JANEIRO - 1977

(14) NICK E. E CABRAL A.
DICIONÁRIO TÉCNICO DE PSICOLOGIA
EDITORA CULTRIX
SÃO PAULO - 1979

(15) PIAGET J. E INHELDER B.
GÊNESE DAS ESTRUTURAS LÓGICAS ELEMENTARES
ZAHAR EDITORES/MEC
RIO DE JANEIRO - 1975

(16) PIAGET J, BETH W.E., DIEUDONNÉ J., LICHNEROUWICZ A.,
CHOQUET G., GATTEGNO C.
L'ENSEIGNEMENT DES MATHÉMATIQUES
DELACHAUX ET NIESTLÉ S.A.
Neuchatel - 1955

(17) PIAGET J.
BIOLOGIA E CONHECIMENTOS
EDITORA VOZES LTDA.
PETRÒPOLIS - 1973

(18) PIAGET J. E INHELDER B.
DA LÓGICA DA CRIANÇA À LÓGICA DO ADOLESCENTE
LIVRARIA PIONEIRA EDITORA
SÃO PAULO - 1976

(19) RODRIGUES M.S.
DICIONÁRIO BRASILEIRO DE ESTATÍSTICA
IBGE
RIO DE JANEIRO - 1970

(20) SAUSSURE F.
COURS DE LINGUISTIQUE GÉNÉRALE
PAYOT
PARIS - 1957

(21) THORNDIKE R.L.
EDUCATIONAL MEASUREMENT
AMERICAN COUNCIL ON EDUCATION
WASHINGTON D.C. - 1971

LÓGICA MATEMÁTICA APLICADA NA COMPUTAÇÃO

Emmanuel P. Lopes Passos, D.Sc.

Introdução

A teoria da computação (leia-se Lógica Matemática) precedeu o surgimento do computador. Lógicos matemáticos já tinham as suas máquinas abstratas antes que a tecnologia fosse capaz de concretizá-las.

Por volta de 1930 Turing criou um modelo rudimentar de computador teórico, composto de dois registros e um conjunto de instruções e que era capaz' de armazenar programas. Esse computador foi chamado' de Máquina de Turing. Ele teoricamente resolve qualquer problema que um computador hoje pode resolver.

Von Newman, baseando-se nesta máquina teórica, criou a arquitetura dos computadores até hoje ' existente (UCP   Memória).

Nesta mesma época, Church criou a teoria ' λ -calculus, que proporcionaria em 1967 ao matemático John McCharty a criação de uma linguagem de programação chamada LISP. Essa linguagem pertence à ' classe das linguagens funcionais, objeto de pesquisa <u>atual</u> na área de Linguagem de Programação.

Markov colaborou na área de computação criando o conceito de <u>Pattern Match</u> que deu origem a ou -

tra linguagem de programação chamada SNOBOL.

Kleene por sua vez influenciou no surgimento do tipo de dados, conceito moderno na Ciência da Computação.

Portanto, o que queremos mostrar com estas' evidencias históricas é que a computação dependeu ' até hoje de Lógicos Matemáticos para sua evolução.' Essa dependencia está na linguagem utilizada na teoria da Computação (lógica Matemática) que permite a descoberta de novos caminhos. Tanto é que os estudos de Turing e Von Newman regem as arquiteturas ' dos computadores de hoje e os resultados de Church' influenciam nas pesquisas para elaboração de um novo computador, para as linguagens funcionais, que ' está sendo pesquisado.

Backus preconiza para o futuro uma arquitetura com funções distribuídas que não fique limitada pelo "gargalo" das arquiteturas de Von Newman(necessidade de tudo passar da memória para a unidade' central de processamento e vice-versa).

A computação (INFORMÁTICA) pode ser pensada de uma maneira <u>macro</u> em tres grandes áreas:

(i) Sistemas de Computação, onde são desenvolvidos' estudos dos computadores existentes e propostas novas linguagens (Fortran, Cobol...), sistemas operacionais, compiladores, etc. (perto da máquina).

(ii) Sistemas de Informações, como Banco de Dados , sistema de recuperação de informações, folha de pagamento, etc. (mais longe da máquina)

(iii) Teoria da Computação - Formalização da Teoria da Computação, Inteligência Artificial, que é a simulação na máquina do comportamento humano (bem longe da máquina).

Dentro da Inteligência Artificial temos uma linha que estuda Prova Automática de Teoremas. É nessa linha que daremos um pequeno exemplo de pesquisa e influência da Lógica Matemática.

OBJETIVOS

Os experimentos realizados pelo autor nos últimos dez anos mostraram em resumo, que:

i) as limitações de tempo e memória nos obrigam a sempre organizarmos um provador interativo homem/máquina.

ii) devemos mudar a estratégia de provas de acordo com a teoria que estamos trabalhando. Pelo fato de ser um sistema em módulos independente, isso é possível em nosso sistema.

Baseados em (i) e (ii), elaboramos um sistema, gerador de provador de teoremas, completamente modular e que ensina ao usuário modificá-lo, isto é, de acordo com resultados parciais de provas naquela teoria, o usuário pode mudar rapidamente o provador.

Basicamente este sistema consta de dois programas modulares. O primeiro programa com que o usuário tem contato, programa gerador, proporciona a geração de um segundo programa gerado, que será o provador de teoremas da teoria definida pelo usuário, quando da interação com o programa.

1- PESQUISAS INICIAIS

A história moderna de demonstradores automáticos de teoremas começa com trabalhos de ROBINSON[1,2,3], que elaborou o Princípio da Resolução, cuja principal vantagem consiste na sua habilidade de evitar maiores obstáculos combinatórios, para ser eficiente.

Seguindo a linha de LINS[4] foram desenvolvidas diversas pesquisas em que o autor desse trabalho participou:

a) Provador para (BAC).
b) Construção de Modelos Minimais no Universo de Herbrand
c) Analisador Sintático para Prova Automática de Teoremas
d) Manipulador de Teorias Definicionais
e) Provador de Teoremas para Point Set Topology e outras, encontradas em PASSOS[5,6,7,8,9,10].

O provador em (a) tinha como objetivo a eficiência em tempo de execução, o que foi conseguido. Foi programado de uma forma linear, sem nenhum tipo de estruturação. O sistema era composto de duas partes distintas: tradução e refutação. A tradução era feita considerando-se $(BAC)_i$ munido de uma estrutura definicional e procurava-se através de uma função de tradução, reduzir todos os símbolos para predicados e funções definidas nas cláusulas, para um único símbolo predicativo c.

O provador contido em (b) era um pouco mais estruturado mas tinha pré-fixado no seu escopo, símbolos funcionais constantes, símbolos predicativos e variáveis da teoria. Quando trocávamos a teoria tinhamos que mudar praticamente todo o programa, para especificar os símbolos novamente. Para termos um programa geral, teríamos que ter infinitos símbolos no programa.

O provador do ítem (c) já deixava o usuário entrar com as variáveis e símbolos e formava a partir daí um analisador' sintático a fim de que pequenos erros na entrada do teorema a ser provado não gerassem provas descabidas.

Finalmente a pesquisa para um provador para "Point Set Topology".

Após todo esse trabalho inicial, o autor optou por uma solução que resolvesse esses problemas e fosse geral. Daí surgiu a idéia do Gerador de Provador que passamos a explicar.

2- GERADOR DE PROVADOR DE TEOREMAS

2-1- IDÉIA GERAL

É a proposta de um SGPT (Sistema Gerador de Provador) ' que basicamente consta de idéias sobre um programa que modifica outro programa, com recuperação de memória.

Esquematicamente existe um programa no nível do usuário, isto é, o usuário tem contato com esse programa. É escrito numa linguagem projetada, especificamente para esse fim, que ' consta de elementos do tipo: declaração da linguagem de 1ª ordem a ser usada, axiomas e definições da teoria dessa linguagem. Funciona como uma declaração em programação, onde dados' seriam as constantes, símbolos funcionais, símbolos predicativos, etc.

O programa gerador monta tabelas, e os "patterns" para o analisador sintático. Este programa modificador de programa, vai editando um outro programa, com os dados inseridos pelo usuário.

Portanto, estamos gerando um programa, para uma determinada teoria, a partir de um programa modificador.

O programa gerado vai constar de definições das tabelas, definição da linguagem, isto é, os "patterns" em SNOBOL que definem, símbolos funcionais, símbolos predicativos etc.

Esta é a primeira parte do trabalho, ou seja, a criação de um gerador (ou editor) automático de programas.

A segunda fase é gerar, a partir das definições, um ' grafo definicional, isto é, monta-se uma matriz que mostra a relação entre as definições. Na apresentação do problema, discutiremos como construir essa matriz. O autor não conhece nenhuma pesquisa que usou essa idéia e provou que funciona, em ' PASSSOS[10].

A importância da criação dessa estrutura definicional' vai aparecer quando se estuda a estratégia de prova por níveis. Passa-se para o nível seguinte no grafo, quando não se conseguir provar o teorema naquele nível. Também o autor provou a completeza dessa estratégia.

Na terceira parte, o programa gerador (compilador) vai montar definicional de tradução. Essa tabela de tradução começaria numa "nodo" escolhida do grafo. Então o programa faria: Crie TRADUZ a partir do nodo X, (a partir da definiçãoX).

Daremos exemplos em "Point Set Topology".

A quarta parte, será feita também pelo usuário, são de laraçōes de que estratégias serão montadas no programa gerado. Esses módulos de estratégias, como conjunto suporte, linear, predicados de comunicação (CP), etc., estão armazenados em algum arquivo independente. O montador (gerador de programas que provam teoremas) recupera nos arquivos esses módulos e monta (edita) no programa que está sendo gerado.

Essas estratégias farão parte do novo programa, assim como os "patterns" já gerados.

Como pesquisa paralela, tem-se alunos construindo novas estratégias, modulares, construídas para não haver conflitos de nomes, parâmetros, etc. Pretende-se ter nesses arquivos (biblioteca de estratégias), todas as estratégias existentes na bibliografia.

2-2 CONTRIBUIÇÃO DA PESQUISA AQUI DESCRITA

A interação homem/máquina proporciona a geração do provador semi automático de teoremas, da Teoria definida pelo usuário.

Esse sistema gerador de provador contribuirá para o pesquisador em prova automática de teoremas em vários aspectos: o primeiro é a de ser um laboratório de prova automática de teoremas em teorias definicionais, onde o matemático poderá conhecer melhor a teoria; o segundo de onde o pesquisador em prova automática de teoremas poderá estudar o comportamento dos provadores gerados e tentar construir um, o mais automáti co possível (mais rápido, menor quantidade de memória utilizada) que poderiam ser utilizados em computadores pequenos.

2.3 - ASPECTOS NA TEORIA DO CONHECIMENTO

Com o sistema funcionando, o pesquisador ao utilizá-lo para provar teoremas, poderá descobrir o que é "inteligente" (naquela teoria). Em outras palavras, se o sistema segue um caminho e esgota quase todas as possibilidades, o usuário define outro caminho e novamente esgota os limites, isto é, o usuário deixa para o sistema a parte "braçal" até descobrir' o melhor caminho, poderemos, com o uso do sistema, descobrir o que é, realmente, difícil naquela teoria matemática. Isto' do ponto de vista da teoria do conhecimento é muito importante.

3 - RESOLUÇÃO

3.1 - PRINCÍPIO DA RESOLUÇÃO

Pode-se consultar CHANG[12] para uma discussão completa de RESOLUÇÃO. Faremos aqui uma breve descrição.

Existem três problemas básicos envolvidos ao suprirmos ' computadores de capacidades dedutivas:

(i) acharmos representações "adequadas" para fatos e relações;

(ii) acharmos regras de inferências "adequadas" para manipularmos esses fatos e essas relações, e

(iii) aprimorar as regras para produzir programas eficientes que possam achar provas numa quantidade razoável de tempo e memória.

Com representação adequada usamos a lógica matemática MARGARIS[12] KREISEL[13], SUPPES[14], WANG[15], como regra de inferência usamos RESOLUÇÃO ROBINSON[2] e finalmente para aprimorar as regras usamos "estratégias", isto é, o refinamento de RESOLUÇÃO. É nessa terceira parte que os pesquisadores ainda utilizam seus esforços.

4- SISTEMA GERADOR DE PROVADOR

4.1-APRESENTAÇÃO DO PROBLEMA

Toda pesquisa desenvolvida pelo autor teve sempre o objetivo de construir um provador automático de teoremas. Veja parte 1.

Com as dificuldades apresentadas no resumo e na parte 1, modificou-se a direção da pesquisa para se buscar um sistema semi-automático. Analisando-se as pesquisas anteriores viu-se da necessidade de criar-se um provador que não dependesse da teoria a ser testada. Daí a solução de criar-se um programa gerador de "provador de teoremas", onde somente as constantes, predicados, símbolos funcionais de uma determinada teoria fossem declarados no início, juntamente com as estratégias que já se encontravam armazenadas, daí então gerar um "provador de teoremas".

Assim, com a experiência adquirida nos sistemas descritos, o autor desenhou um sistema geral para prova de teore mas, sem esses obstáculos citados anteriormente e com a ótica de um pesquisador em prova automática de teoremas.

A modularidade do sistema era importante, pois queria-se construir estratégias ao longo do tempo, e acrescentá-las ao sistema, sem modificar sua lógica.

4.2 - SOLUÇÃO DE FORMA ESQUEMÁTICA

A solução foi um módulo conversacional a nível de usuário, onde o usuário declara a linguagem de 1ª ordem que usará naquela teoria, entra com os axiomas e definições da teoria, seleciona estratégias, e então esse sistema gera um "provador de teoremas" para aquela teoria, veja figuras 1 e 2.

Este provador também produz um histórico de todos os "caminhos de prova" para a teoria em questão.

ESQUEMA DO SISTEMA GERADOR DE PROVADORES DE TEOREMAS - SGPT

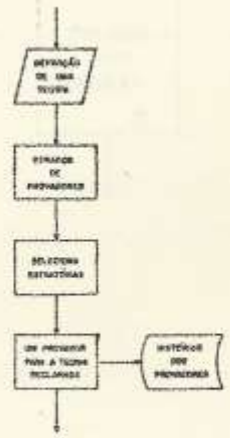

FIG. I

ESQUEMA DO PROVADOR GERADO

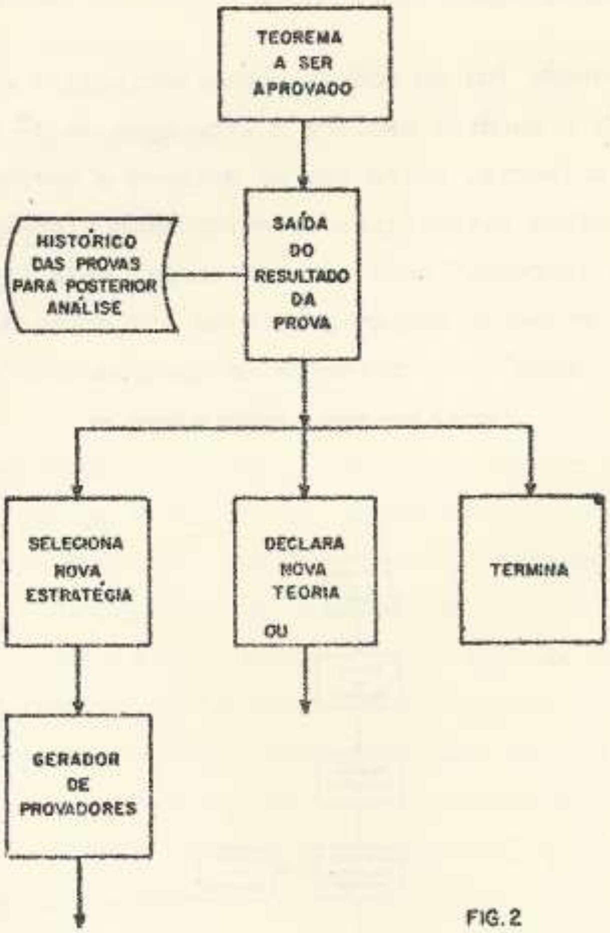

FIG. 2

4.3 - DESCRIÇÃO DO SISTEMA

4.3.1- CARACTERÍSTICAS DO SISTEMA GERADOR DE PROVADOR DE TEOREMAS - SGPT

O SGPT foi desenvolvido com o objetivo de pesquisar, ' em prova automática de teoremas, a melhor maneira de construir-se provadores inteligentes nas teorias estudadas.

É composto basicamente de dois programas. O programa ' que o usuário tem contacto, que é conversacional, onde ele só tem de responder as perguntas para que seja gerado automaticamente e o programa que irá prover teoremas. Estes programas ' são modulares com partes independentes. As estratégias ficam' em locais fixos e pode-se acrescentar novas estratégias nesses locais para uma melhor escolha por parte dos usuários. ' Elas podem ser construídas por pessoas diferentes.

4.3.2 - PROGRAMA GERADOR

É composto de uma parte estática onde o usuário fornece dados para a geração do "provador" e uma parte dinâmica na qual o usuário interage com o programa para determinar ou modificar as combinações das estratégias e buscar dentre as estratégias existentes, aquelas selecionadas, por intermédio de comandos, pelo usuário.

A linguagem na qual escreve-se este programa, será projetada, completamente, numa próxima pesquisa. Aqui tem-se somente um protótipo da linguagem, com o qual testa-se o sistema que consta de:

DECLARAÇÕES - da linguagem de $1^{\underline{a}}$ ordem a ser usada
AXIOMAS - da teoria dessa linguagem
DEFINIÇÕES - dessa mesma teoria
COMANDOS - para manipular as estratégias e principalmente para editar (gerar) o provador.

Este programa gerador contém, no seu módulo principal, o compilador dessa linguagem descrita acima. A função desse compilador é: (i) montar as tabelas usadas nas estratégias de Resolução, isto é, edita as tabelas usadas que comporão o texto do programa gerado. (ii) definir os "patterns" do programa gerado, ou seja, os padrões dos símbolos funcionais, símbolos predicativos e das constantes para o analisador sintático. (iii) gerar as tabelas de traduções (começa numa folha do grafo que define a estrutura definicional da linguagem) que farão parte do programa gerado e auxiliarão na estratégia de traduzir por níveis.

4.3.3 - PROGRAMA GERADO

Programa construído pelo programa gerador por intermédio dos dados inseridos pelo usuário. Consta de:
os "patterns" em SNOBOL que definem os símbolos predicativos, símbolos funcionais e as constantes; tabelas de traduções; estratégias que serão usuadas nas provas dos teoremas: comandos que permitam ao usuário voltar ao programa gerador e redefinir o programa gerado, mudando-se as estratégias ou combinações dessas; comandos que permitam mudar a maneira da busca do espaço solução; módulos de leitura e impressão, para as cláusulas

que definem o teorema a ser provado; e finalmente o analisador sintático.

Haverá comandos no programa gerado, que permitirão um tipo de recuperação de memória, ou seja, será possível ao sistema retroreferenciar e guardar somente as cláusulas úteis, ' eliminando da memória as que não servirão para gerar cláusulas úteis. A lógica do programa gerado será dada a seguir.

O programa gerador e o gerado compõemo SGPT que fundamentalmente tem três níveis de ação:

(1) gera um programa

(2) modifica programa

(3) modifica espaço de memória obtido por aquele programa gerado, isto é, dirige a busca do espaço solução.

4.3.4 - FLUXO DA DEFINIÇÃO DE UMA TEORIA

A definição da Teoria é feita em duas etapas:

figura 3

Etapa 1 - O usuário define as CONSTANTES, VARIÁVEIS' PREDICADOS E FUNÇÕES.

O SGPT transforma as listas fornecidas pelo usuário ' em "patterns" e "tables", armazenando-os no arquivo ALFABETO.

Por exemplo:

a - Entre com as CONSTANTES da Teoria

A,B,C $

O SGPT monta o "pattern" das constantes

SCONST = SCONT / SCONST NUMNT / NUMNT

onde o "pattern" NUMNT = SPAN ('0123456789').

O significado dessa declaração de constantes é que qualquer constante da Teoria é a permitida pelo "pattern" SCONST.

b - Entre com as VARIÁVEIS da Teoria

X,Y,Z $

O SGPT monta o "pattern" das variáveis

SVAR = SVAR / SVAR NUMNT

O significado é que qualquer variável da teoria é a permitida pelo "pattern" SVAR.

ESQUEMA DO FUNCIONAMENTO DAS DUAS PRIMEIRAS ETAPAS

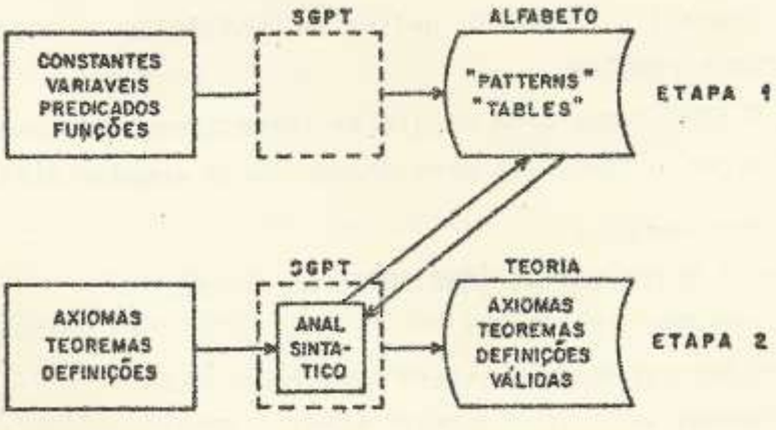

FIG.3

c- Entre com os PREDICADOS OU FUNÇÕES

F, 2 ; I, 1 $ dois nomes de funções com aridades 2 e 1

P, 2 ; = , 2 $ dois símbolos de predicados com arida - des 2 e 2.

4.3.5 - FLUXO DO FUNCIONAMENTO DO COMANDO AXIOMA

figura 4

Os comandos AXIOMA, TEOREMA e DEFINIÇÃO têm o mesmo ' funcionamento (veja no Capítulo quatro a explicação teórica).

A ação do SGPT PARA ESSES COMANDOS é dividida em duas partes:

(i) "Monta", internamente, o "programa" que contem o código-SNOBOL dos arquivos ALFABETO e ANALISADOR, usando a função CODE.

(ii) Coloca as iniciais AX na tela e espera o usuário' escrever uma linha. Quando isto é feito, o SGPT entrega a linha ao programa descrito em (i). Este programa aceita ou rejeita (envia uma mensagem à tela) essa linha, repetindo (ii) até o usuário escreve uma linha começando com a palavra 'FIM'.

4.3.6 - ANALISADOR SINTÁTICO

Este programa auxilia ao usuário analisar se sua entrada de FUNÇÕES, PREDICADOS, AXIOMAS (TEOREMAS, DEFINIÇÕES) estão sintaticamente corretas.

Por exemplo:

No nosso caso definimos a função F com aridade 2 e não definimos W como variável.

ESQUEMA DO FUNCIONAMENTO DO COMANDO AXIOMA

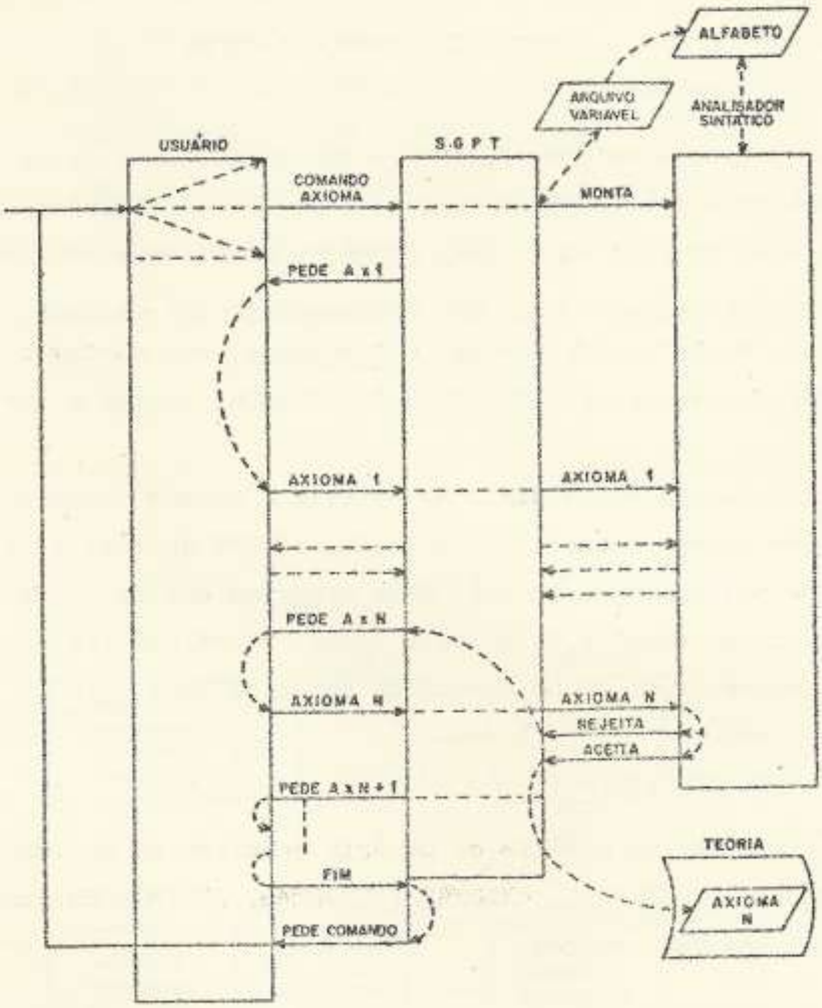

FIG. 4

Se dermos entrada na função F(W) o analisador rejeitará pois a aridade da função F é 2 e aí só temos 1 variável e também W não é uma variável de nossa teoria.

Nesse momento, depois de executadas as etapas 1 e 2 temos: Um arquivo com os "patterns" e "tables" das constantes variáveis, predicados e funções - (ALFABETO).

Um arquivo com os AXIOMAS, TEOREMAS e DEFINÇÕES válidas- (TEORIA).

Um arquivo com as ESTRATÉGIAS prédefinidas (ESTRATÉGIAS).

4.3.7 - FLUXO DA MONTAGEM DAS ESTRATÉGIAS

O SGPT conterá uma linguagem de consulta para montagem e recuperação de estratégias que serão compostas nesse primeiro ' ano, de sete estratégias. A saber:

1) Eliminação (subassunção de cláusulas e tautologias)
2) Preferência mais curta
3) Conjunto Suporte
4) Linear
5) Hiperesolução (caso especial de resolução)
6) Lock Resolution (indexação de predicados)
7) Paramodulation

Por meio da linguagem de consulta para estratégias existente no SGPT, o usuário monta no arquivo TEORIA as estratégias que o provador de teoremas usará nas provas de teoremas. Ver figura 5.

FIGURA 5

4.3.8 - FLUXO DO FUNCIONAMENTO DO COMANDO EXECUTE PROGRAMA
Figura 6

A ação do SGPT, quando executamos o comando EXECUTE , ver figura 6.

4.3.9 - FLUXO DO FUNCIONAMENTO DO COMANDO EXECUTE (INTEGRADO)
Figura 7

Sintaxe: EXECUTE arquivo

O objetivo desse comando é permitir ao usuário determinar a execução de um programa em código-SNOBOL contido no "arquivo" especificado.

A utilização por ora é:
EXECUTE PASSADO ou
EXECUTE HISTÓRIA

A ação do comando é recuperar tudo o que foi feito na sessão sem precisar refazer nada, isso se a sessão, eventualmente tiver sido cancelada.

5- CONCLUSÃO

O que nós fizemos até agora foi estudar e construir provadore de teoremas para alguma classe, de teorias axiomáticas, que chamaremos Teorias Definicionais. Essas Teorias são aquelas que contém um grande número de definições. "Point Set Topology" é um exemplo de ' tal teoria.

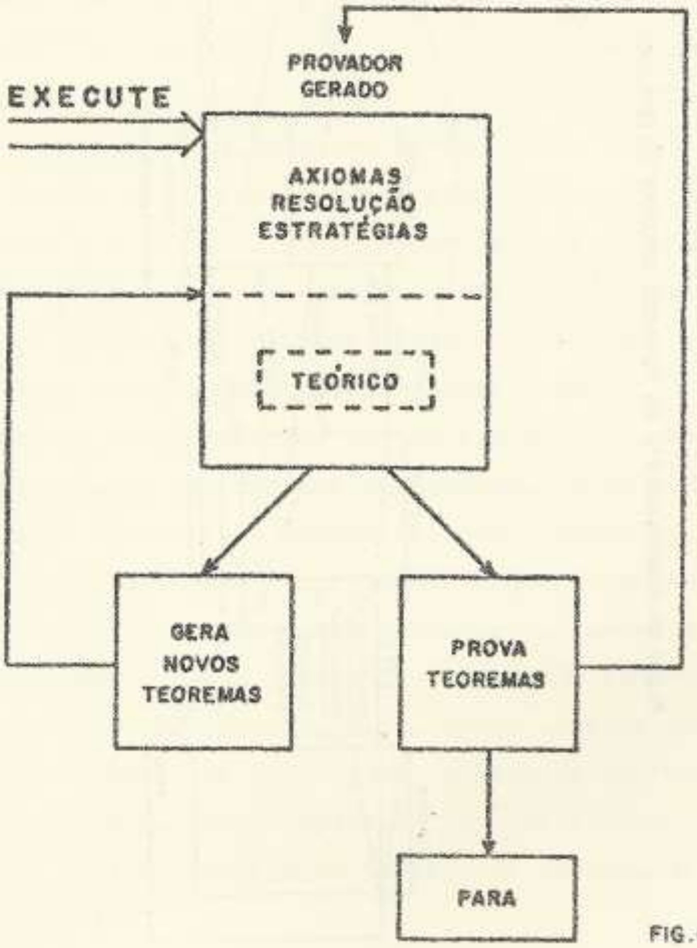

FIG.6

FLUXO DO FUNCIONAMENTO DO COMANDO EXECUTE (INTEGRADO)

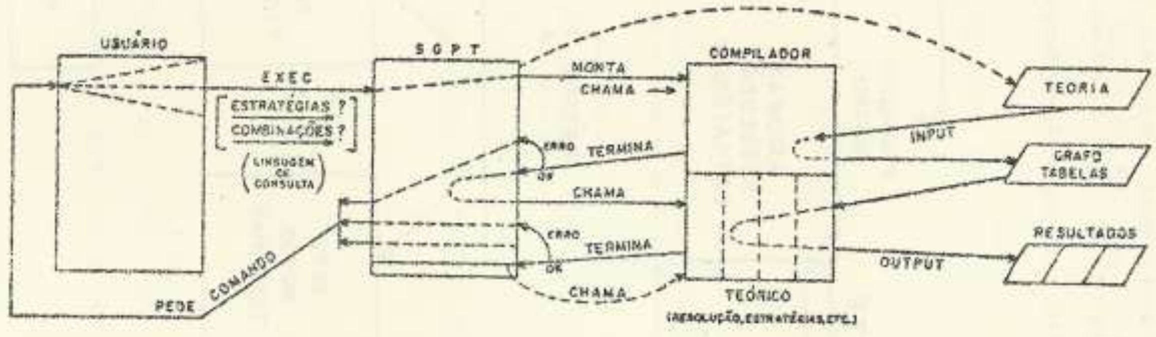

FIG. 7

Sabemos que um determinado conjunto de sentenças, que ' constitui uma teoria, admite vários sistemas axiomáticos, isto é, há diferentes partições possíveis dessas sentenças em ' axiomas e teoremas, e de seus objetos (termos e predicados em objetos primitivos e definidos).

Dada uma teoria T que admite vários sistemas axiomáticos, é muito difícil decidir qual sistema usaremos para construir o provador de teoremas para T. É muito possível que a eficiência desse provador seja uma função dessa escolha, por isso, devemos testar os vários sistemas axiomáticos para então decidirmos qual é o melhor. Isto o nosso gerador de provador proporciona. Mas fica existindo, ainda, a dificuldade de encontrar na literatura disponível, axiomatizações diferentes, para mesma ' teoria, para termos uma completa formalização.

Uma vez encontrada uma axiomatização, nós devemos verificar suas conexões com teorias conhecidas. Essas conexões são, em geral, muito difíceis de detetar se não conhecemos bem essa nova teoria. Portanto nos resta tentar conhecê-la bem, por intermédio do conhecimento de suas diferentes axiomatizações.

O sistema por nós proposto e explicado tem como objetivo facilitar, interativamente, que o usuário aprenda sobre a Teoria, mudando os axiomas, as definições, e gerando teoremas para se conhecê-la melhor, pela comparação dos resultados. Aí, então, ' decidir de qual axiomatização deverá ser construído o provador' de teoremas final.

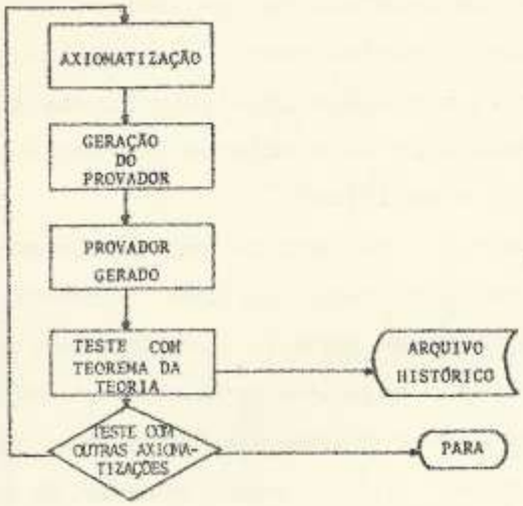

FIG. 8

O objetivo inicial era seguir o caminho deixado pelo trabalho de LINS[4] para Demonstração Automática de Teoremas em "Point Set Topology". Em LINS[4] já existiam algumas idéias que foram realizadas ad hoc e sumariamente. O autor iniciou os estudos e realizou, então, várias experiências, que foram citados na parte um deste trabalho.

Como decorrência desse estudo, verificamos a necessidade de criarmos um sistema geral para teorias definicionais.

Foi então que propusemos o SGPT.

REFERÊNCIAS BIBLIOGRÁFICAS

1. ROBINSON, J.A. - THEOREM PROVING ON THE COMPUTER - J. ACM 10, 1963.
2. ROBINSON, J.A - A machine-oriented logic based on the resolution principle - J.ACM 12, 1965.
3. ROBINSON, J.A - A review of automatic theorem proving- Proceedings of Symposia in applied mathematics volume 19, 1967.
4. PASSOS, E.P.L., de CARVALHO, R.L. e PEIXOTO, S.R. - Communication Predicates: Complete Strategy for Resolution-Based Theorem-Provers An Evaluation of An Implementation - Current Topics in Cybernetics and Systems WOGSC,1978, Springer-Verlag.
5. PION, M.M.B., de CARVALHO, R.L. e PASSOS, E.P.L. - Interactive System to Construct Minimal Models on the Herbrand Universe, Publ. 8, IME, 1980.
6. LANZELOTE, R.S.G., PASSOS, E.P.L., de CARVALHO, R.L. - MTD a Conversational System Oriented for Topology, ENCR-80, Austria.
7. PASSOS, E.P.L. and SILVEIRA, G.G - Quantifier Elimination to Mechanical Theorem Proving - Implementation - works from 3th Intern. Cong. of Cybernetics and System August 1975, Bucarest - Romenia, Publicado pela NOSTH-HOLLAND em 1976.
8. PASSOS, E.P.L. - Introduction to Mechanical Theorem Proving - Monographs in Computer Science and Computer Applications' 2, 1971, INF-PUC/RJ.

9. PASSOS, E.P.L. - Sistema Auxiliar do Matemático (S.A.M) - Anais do 6º Painel de Discussão sobre Tópicos de Computação, 1979, Valparaiso - Universidade Católica - Chile.

10. PASSOS, E.P.L. - Algumas Idéias e Experimentos sobre Demonstração Automática de Teoremas - Tese de Doutorado Sistemas - COPPE/UFRJ, 1981.

11. CHANG, C.L. and LEE, R.C.T. - Symbolic Logic and Mechanical Theorem Proving - New York, Academic Press, 1973.

12. MARGARIS, A. - First Order Mathematical Logic - Waltham, Mass., 1967.

13. KREISEL, G. and Krivine, J.L. - Elements of Mathematical Logic: Model Theory - Amsterdam, 1967.

14. SUPPES, P. - Introduction to Logic - Prentice-Hall, N.Y., 1957.

15. Wang, H - Logic of Many Sorted Theories - JSL 17, 1952.

EDUCAÇÃO MATEMÁTICA NA ESCOLA ELEMENTAR

Recursos Utilizados na Pesquisa Experimental "Binômio Professor-Aluno"[1]

Moema Sá Carvalho

"A escola primária tem de ser a mais
importante do Brasil depois a média,
depois a escola superior."

Anísio Texeira (1958).

Uma iniciação elementar mal orientada na aprendizagem da Matemática é responsável por muitos dos futuros fracassos ao longo do 2º e do 3º grau. E a má iniciação em Matemática vem se tornando cada vez mais frequente em nossas escolas primárias.

---

[1]No Boletim 10 do GEPEM encontra-se relato geral dessa pesquisa experimental que contou com o apoio técnico e financeiro do MEC-INEP, e com a equipe de professores do GEPEM:

Pesquisadores
Ana Lúcia Bordeaux
Cristina Spinola Caldas
Maria José Montes
Vera Maria Rodrigues

Coordenador:
Anna Averbuch

Supervisores
Estela Fainguelernt
Franca C. Gottlieb
Moema Sá Carvalho

Coordenador Geral:
Maria Laura L.Lopes

A conclusão da pesquisa experimental "Binômio Professor-Aluno" reforça essas afirmações, ao apontar falhas no ensino da Matemática nos níveis elementar e de formação do magistério primário.

Durante essa pesquisa constatou-se que estava sendo desprezada a filosofia pragmática concebida por John Dewey (1859 - 1952) e aqui preconizada desde os anos 20, como norteadora da "Escola Nova", por Anísio Texeira (1900 - 1971), Fernando de Azevedo (1894 - 1974), Lourenço Filho (1897 - 1970), entre outros.

"Em Dewey é central o conceito de experiência pois a ele se liga o problema do conhecimento. (...) São inseparáveis vida, experiência e aprendizagem. Reconstruir a experiência, melhorar o nível de sua qualidade, será portanto papel da educação, desse modo não limitada à aquisição de resultados educativos por meio de puros processos verbais.

De fato, não bastam palavras. Educar no contato direto com as crianças exige ambiente propício à atividade de agir, pensar e sentir do educando. A escola,portanto,deve ser conduzida como lugar em que se prepara para viver".[2]

---

[2] Hermes Lima, em "Anísio Teixeira Estadista da Educação", Editora Civilização Brasileira, 1978:

Concorriam para esse procedimento a ausência de uma filosofia educacional, a falta de estudos sobre psicologia da aprendizagem, além do despreparo' das professoras no setor específico da Matemática.

Felizmente constatou-se que se pode abrir caminhos em que professoras primárias em exercício' melhorem o seu desempenho, conseguindo recuperar ' seus alunos para um bom aprendizado.

Nessa pesquisa experimental foi fornecido embasamento matemático às professoras, visando, principalmente, obter certos progressos nos métodos escolares voltados para a formação da criança. Esses ' enfoques de formação, é bem diverso do que visa apenas adestramento em habilidades específicas e que ' constitui uma tentação, até compreensível, se bem ' que condenável, em face das tentativas frustradas de um bom procedimento educacional.

No processo ensino - aprendizagem de Matemática são efetivamente delicados os momentos de assimilação de conceito e os de passagem para o uso da linguagem matemática.

Esses momentos, se bem conduzidos, respeitando o rítmo e formação das estruturas mentais de cada um, consolidam as bases de formação na criança' de seus processos de matematização.

O recurso ao uso do material concreto, o manuseio desse material, a vivência infantil devem constituir o apoio e o estímulo para a formação dos conceitos matemáticos elementares, como o da contagem ou o das operações aritméticas.

Por sua vez, a assimilação dos conceitos deve preparar a introdução gradativa da linguagem matemática, na sua simbologia.

Se, ao contrário, esses momentos delicados, porém importantes, forem desrespeitados ou tumultuados no seu encadeiamento, o que se consegue junto aos educandos é que se tornem bons repetidores, capazes de reproduzir por imitação os mecanismos que lhes exibiram, e nada além disso.

Dessa maneira se adestrariam as crianças, mas não se estimularia o desenvolvimento de suas aptidões naturais.

Uma das tônicas da equipe de pesquisadores constituiu em orientar as professoras assistidas de modo a que se sentissem seguras para poder estabelecer como um recurso didático normal que: Atividades dos seus alunos, executadas ao vivo, deveriam preceder o registro no papel do fato matemático em foco.

Tomava-se o cuidado de esclarecer que essas atividades não deveriam encobrir ou dissimular os fatos matemáticos - ao contrário, deveriam proporcionar oportunidades de "redescobertas" através de vivências individuais, devidamente orientadas para a aquisição do conceito em vista. Outras atividades poderiam ser programadas depois, com objetivos de fixação.

É importante que as professoras primárias saibam distinguir umas das outras, nos seus objetivos diversos.

Pretendemos neste artigo apenas comentar alguns exemplos de situações de dificuldades encontradas no início da experiência e como foram superadas com os simples recursos a que nos referimos:

I. Dificuldades na Utilização e no Cálculo das Operações Aritméticas:

I.I Adição e Subtração (1º e 2º séries).

I.2 Multiplicação e Divisão (3º e 4ª séries).

II. Dificuldades na Representação dos Números (1ª e 2ª séries)

III. Dificuldades e Desinteresse quanto ao Uso e à Compreensão das Unidades de Medida (4ª série)

Passemos a comentá-las.

I. Dificuldades na Utilização e no Cálculo das Operações Aritméticas.

Os alunos que apresentavam as dificuldades indicadas em ' I.I e I.2 não haviam ainda dominado o significado das operações aritméticas para as quais estavam sendo solicitados, exigindo-lhes o registro do algoritmo por imitação do modelo apresentado.

A antecipação do simbolismo, galgando etapas, provocava ' confusões em meninos que estavam sendo compelidos a utilizar' uma linguagem simbólica cujo significado lhes escapava.

O recurso ao simbolismo da linguagem matemática havia precedido a compreensão do seu significado, quer nas 1ª série e na 2ª, quer na 3ª e 4ª. Consequentemente, as crianças não se situavam bem ante os problemas propostos. "Que conta tenho que fazer?" Seria pergunta bem característica dessa situação.

Visando a formação no educando dos conceitos das operações aritméticas, foi sugerido o manuseio de coleções (ou "caixas de contagem") com pequeno número de objetos, como canudos, ou chapinhas etc, a serem gradativamente utilizados para contagens, adições subtrações, decomposição de um número em parcelas, para a 1ª e 2ª séries, e distribuição em partes iguais, sem resto, ou com resto, para as 3ª e 4ª séries.

Uma vez tendo as crianças dominado cada conceito, sugeriu-se que o registrassem no papel, primeiro desenhando coleções e depois escrevendo as contas, na convenção habitual.

O registro da linguagem simbólica se fez, então, após os alunos terem tido oportunidade de calcular através do manuseio do material concreto. As crianças voltaram assim a se confrontar com as operações, sua linguagem matemática e sua representação simbólica, num processo gradativo, cujos parâmetros, dessa vez, eram as suas próprias estruturas cognitivas.

Mais adiante foram propostas contas diretamente no papel, ficando os alunos com liberdade para recorrer às caixas de contagem, ou aos próprios dedos, se o preferissem, sempre que precisassem. Foi esse um momento de fixação, enriquecido mais adiante, com jogos opcionais.

Problemas simples foram então sugeridos, visando utilização e fixação do que haviam compreendido.

Em complementação, em uma das escolas foi pedido às crianças da $1^{\underline{a}}$ série que expressassem com um desenho a situação descrita nos problemas apresentados. Essa sugestão,por um lado, beneficiou aqueles alunos cuja alfabetização era ainda deficiente, e, por outro lado, permitiu que a assimilação do enunciado se desse de um modo mais completo; estimulou-se ainda, com essa medida, a criatividade das crianças.

Na $3^{\underline{a}}$ e $4^{\underline{a}}$ séries foi sugerido que os alunos:

.. inventassem situações-problemas, contando pequenas histórias que se relacionassem com as contas indicadas' pela professora;

.. mantivessem atividades de troca, venda ou compra, nas próprias turmas.

Paralelamente foi sugerido às professoras que se inteirassem sobre métodos didáticos já experimentados; por exemplo, o relatado pelo Prof.Luiz Alberto Brasil, em "Experiências Pedagógicas Baseadas na Teoria de Piaget" (Forense' Universitária, R.J. 1979).

Resultados obtidos:

Acabaram-se as dificuldades.

As deficiencias encontradas no início da experiência não se repetiram em 1980, entre alunos provenientes de turmas já iniciadas através do manuseio de material concreto, por orientação de pesquisa experimental em apreço.

Provavelmente por que já estivessem mais condicionados à mecânica do algarítmo, sem o domínio do que realmente ' significava a operação, os alunos das $3^{\underline{a}}$ e $4^{\underline{a}}$ série menciona-