

Você foi encarregado de projetar o circuito de dados da hierarquia de memória para um sistema embarcado que executa um conjunto limitado de aplicações. Para tanto, os engenheiros de aplicação lhe entregaram um conjunto de *traces* com as referências a dados dos programas. Cada um dos *traces* contém a sequência de endereços de dados referenciados pelo programa, e um indicador para o tipo de referência, se leitura ou escrita.

Você deve usar o simulador de caches *dinero* para medir a taxa de acertos de cada uma das configurações de cache. Em função da taxa de faltas você deve estimar o tempo médio de acesso à memória de cada configuração e recomendar o melhor projeto aos engenheiros de produto.

Além do tempo médio de acesso à memória, você deverá avaliar a área e o consumo de energia do seu projeto, conforme os dados que você recebeu dos engenheiros de projeto.

Estes artigos podem ajudar na compreensão do que está sendo solicitado:

<http://www.inf.ufpr.br/roberto/wscad09pcvc.pdf>, e

http://www.inf.ufpr.br/pos/techreport/RT_DINF002_2011.pdf.

Etapa 1 Copie os fontes da página da disciplina e o desempacote em um diretório apropriado:

```
wget http://www.inf.ufpr.br/roberto/ci312/trab15pri.tgz
tar xzvf trab15pri.tgz
```

Se não existem, crie os diretórios `$HOME/bin` e `$HOME/man/man1`, compile e instale os executáveis e páginas de manual:

```
mkdir ~/bin
mkdir -p ~/man/man1
cd trab151
(cd dinero ; make ; make install)
(cd tlb ; make ; make install)
(cd sched ; make ; make install)
cp man/*.1 ~/man/man1
```

Se ocorrer algum erro de compilação, por favor avise ao professor.

Edite seu `~/bashrc` e acrescente os novos caminhos às suas variáveis de ambiente, conforme abaixo:

```
export PATH=$PATH:$HOME/bin
export MANPATH=$MANPATH:$HOME/man
```

Etapa 2 Uma vez que o ambiente de simulação esteja instalado, defina os parâmetros de projeto da cache que devem ser investigados e planeje os experimentos. Veja a página de manual de *dinero* para os parâmetros de projeto da cache que podem ser determinados nas simulações.

Veja os *scripts* `dinero/run` e `dinero/LOOP` para ter uma ideia das possibilidades. Cada simulação gera um ponto do espaço de projeto. São necessárias N simulações para verificar N valores de um determinado parâmetro. Cada simulação gera um arquivo com os resultados da simulação e os valores de interesse (taxa de faltas) devem ser extraídos do arquivo e tabulados.

A taxa de faltas obtida de uma simulação depende do *trace* usado naquela simulação porque cada *trace* corresponde a um programa distinto. Os *traces* estão no diretório TRC.

Com os valores tabulados, um gráfico com os resultados pode ser gerado para acelerar a análise e observar tendências (taxa de faltas \times variação de um parâmetro). O *script* `plot.gp` recebe um conjunto de dados e produz um gráfico com o programa `gnuplot`. Veja o enorme conjunto de possibilidades com `man gnuplot`.

Inicialmente, faça uma exploração do espaço de projeto usando somente um *trace*.

Uma vez que um conjunto pequeno de projetos foi escolhido (2–4 projetos), os experimentos devem ser repetidos com estes projetos para todos os *traces*, para garantir que sua escolha é a melhor para todos os aplicativos. Veja, no livro texto, como computar um número para o desempenho do seu projeto que realmente quantifique sua qualidade.

Etapa 3 Uma vez escolhidos poucos projetos para a cache (2–4 projetos), você deve verificar o desempenho do seu projeto com uma carga relativamente realística. Para tanto use o programa `sched` para escalonar a execução dos vários aplicativos na cache e verificar, sob esta carga, qual é o melhor projeto.

Execute `sched/run` para entender o escalonamento da ‘execução’ dos processos. Escolha um intervalo razoável para a troca de processos executando no processador.

Nesta etapa você deve simular uma carga com os vários processos executando no processador e escolher qual o melhor projeto da cache para a(s) carga(s) simuladas. Você pode variar o *quantum* e a mistura de processos, até o limite de 8 processos concorrentes, imposto por `sched`.

Etapa 4 Com base nos dados sobre área e energia, ao final desta especificação, reveja a qualidade das suas escolhas na proposta de projeto de cache. Por exemplo, seu projeto pode atingir uma taxa de acertos excelente, a um custo demasiado elevado em termos de área e/ou energia.

Na avaliação das alternativas de projeto os parâmetros têm os seguintes pesos:

área: 20%

energia: 30%

tempo médio de acesso: 50%

O tempo de acesso do segundo nível é 50ns. Custa 1.0ns para transferir cada palavra do nível 2 para o nível 1.

O tempo de acesso da L1 (t) deve ser depreciado com a expressão abaixo.

$$q(t) = 0.786 + 1.10084t + -1.01865t^2$$

Quanto maior o tempo de acesso, pior é a qualidade q do seu projeto.

Etapa 5 Determine os parâmetros de projeto para uma TLB, a ser instalada entre a cache de dados e a memória. Para tanto use os mesmos procedimentos das etapas anteriores. Os *scripts* `tlb/run0` e `tlb/run2` indicam as possibilidades. O tempo de acesso da TLB não é relevante e não deve ser considerado nesta fase do projeto.

O diretório `plots` contém um *script* e um conjunto de dados para plotar um gráfico de “taxa de faltas \times tamanho dos blocos” usando o programa `gnuplot`. Você deverá adaptar o *script* para mostrar seus resultados. Se você preferir gerar os gráficos com uma planilha, isso também é aceitável.

A escolha dos eixos, “taxa de faltas” e “tamanho de blocos” é adequada para todas as medições? “Capacidade” talvez seja uma abcissa mais interessante.

Especificação:

1. O trabalho pode ser efetuado em duplas;
2. o arquivo com os produtos deve ser nomeado `xx-yy.tgz` sendo `xx` e `yy` os *usernames* dos componentes do grupo, e todos os arquivos relevantes deverão estar abaixo do diretório `xx-yy`;
3. PLÁGIO NÃO SERÁ TOLERADO. É interessante que os alunos conversem sobre o projeto mas cada grupo deve definir sua própria estratégia de projeto;
4. seu trabalho será avaliado por: (i) qualidade global do resultado – um projeto de baixo consumo, área pequena, e bom desempenho; (ii) qualidade e quantidade das simulações – força bruta piora a nota; (iii) qualidade técnica da apresentação dos resultados.

Produtos:

1. Relatório em papel A4, com letras em 11 pontos, espaço simples, formatação simples, contendo os nomes dos componentes do grupo, e as conclusões de projeto, apontando o(s) melhor(e)s projeto(s) e justificando claramente as escolhas. Seu relatório deve conter os gráficos com os resultados das simulações;
2. presença dos membros do grupo na data e hora marcadas para a apresentação.

Sugestões:

1. Assegure-se de que entendeu a especificação antes de iniciar o projeto das caches e as simulações;
2. as simulações são demoradas e geram enormes conjuntos de dados; planeje os experimentos e a coleta e tabulação dos resultados antes de disparar os *scripts* com as simulações.

Área, tempo de acesso, energia por referência

Os engenheiros de projeto computaram os seguintes gráficos, para auxiliar na escolha do melhor projeto da cache de dados.

As capacidade simuladas são 1,4,16 e 64 Kbytes, as associatividades 1,2,4 e 8, e os tamanhos de bloco 8,16,32,64 e 128 bytes.

O primeiro conjunto de valores relaciona capacidade, associatividade e tamanho de bloco com a área resultante do projeto, em mm^2 , e é mostrado na Figura 1.

O segundo conjunto de valores relaciona capacidade, associatividade e tamanho de bloco com o tempo de acesso resultante, em ns, e é mostrado nas Figuras 2 e 3. Note que o tempo de acesso nem sempre cresce linearmente com a capacidade, tal como área ou energia, porque o simulador escolhe a melhor geometria para otimizar o tempo de acesso. Todas as capacidades entre 1 K e 64 Kbytes foram simuladas para o tempo de acesso.

O terceiro conjunto de valores relaciona capacidade, associatividade e tamanho de bloco com a energia dispendida em cada acesso à cache, em nJ, e é mostrado na Figura 4.

O simulador empregado para estas simulações é o CACTI¹.

Histórico das Revisões:

20mar acrescidas restrições quanto a tempo de acesso, área, energia;
25fev primeira versão.

¹CACTI 3.0: An Integrated Cache Timing, Power, and Area Model, Premkishore Shivakumar and Norman P Jouppi, Tech Report WRL-TR-2001.2, aug 2001, Compaq Western Research Lab.

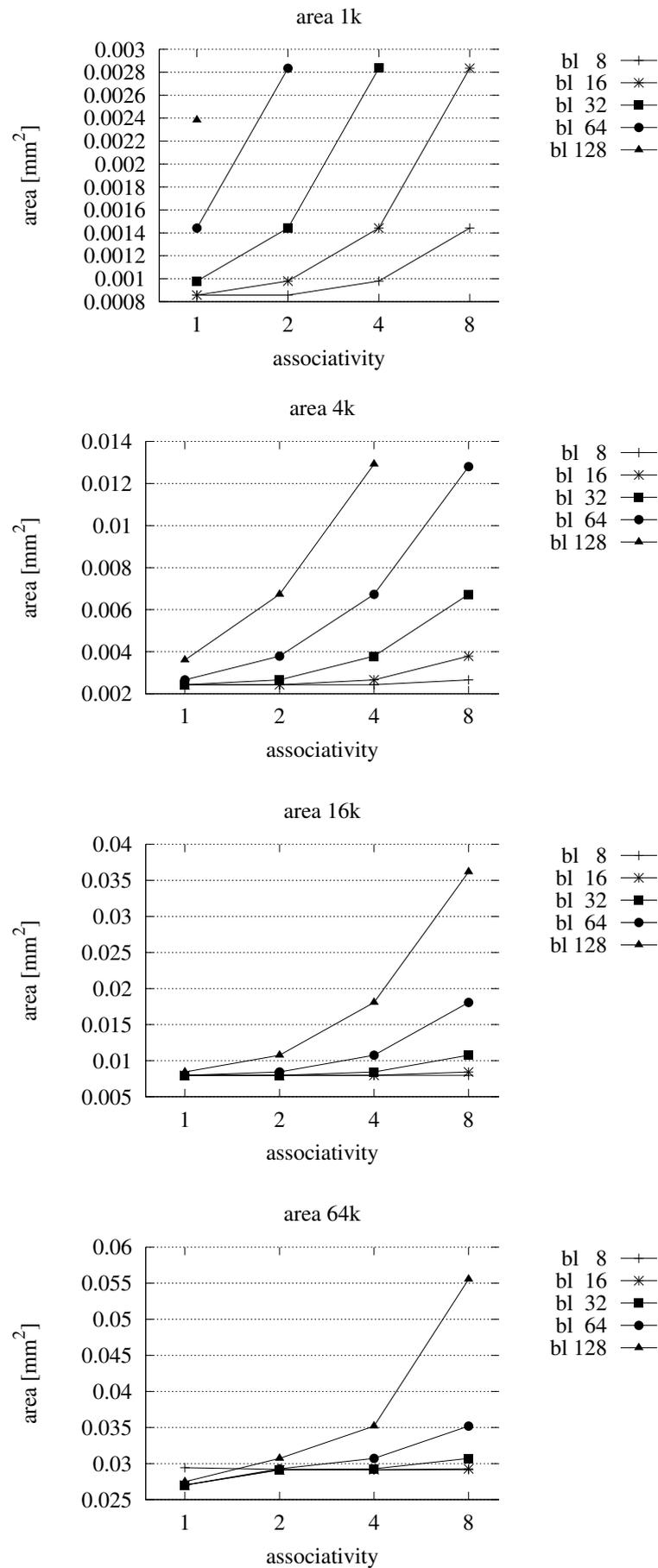


Figura 1: Área X capacidade X associatividade X tamanho de bloco.

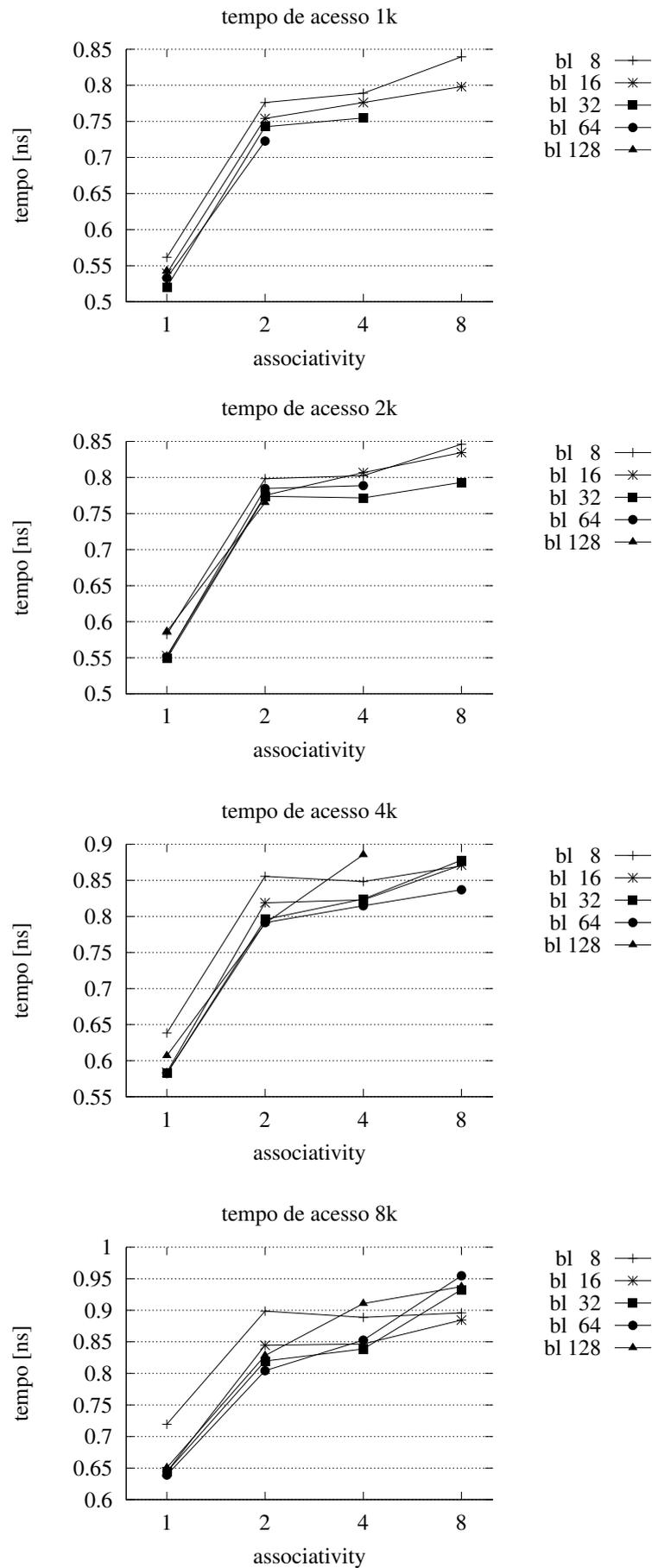


Figura 2: Tempo de acesso X capacidade X associatividade X tamanho de bloco.

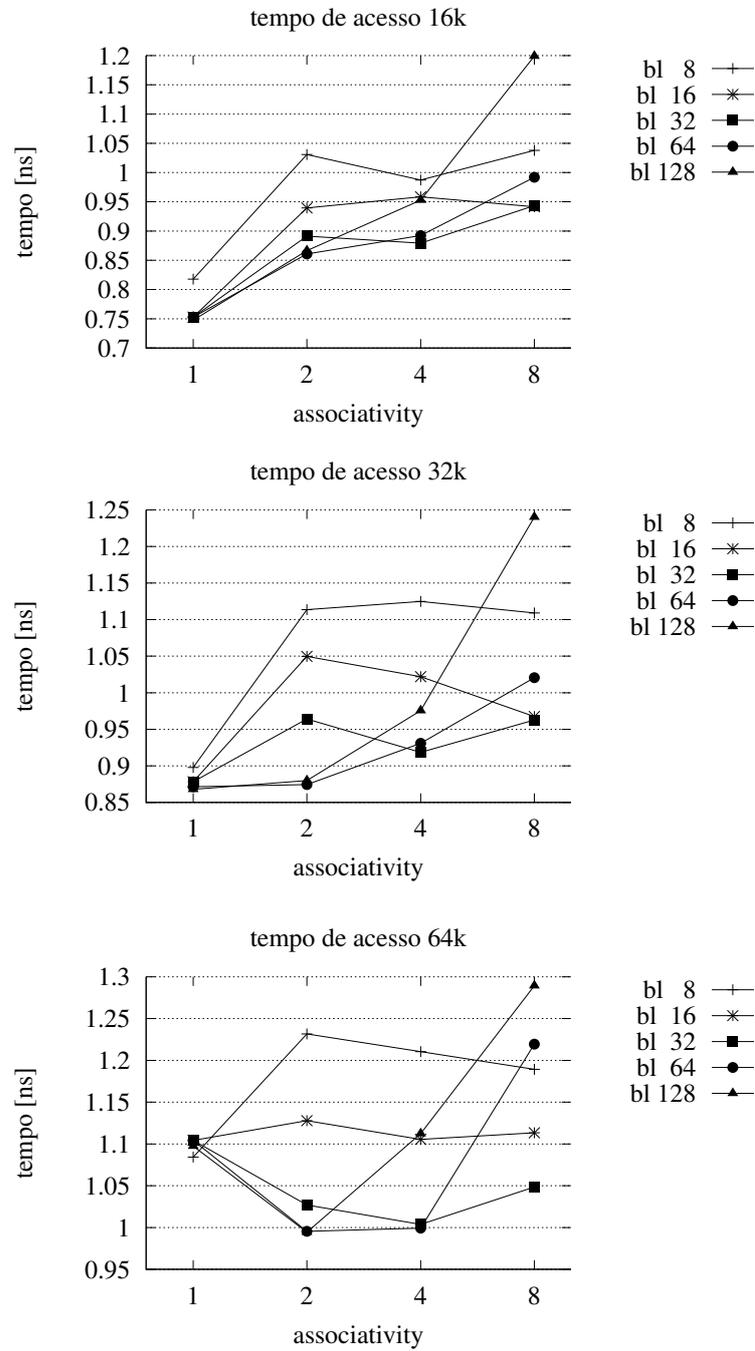


Figura 3: Tempo de acesso X capacidade X associatividade X tamanho de bloco.

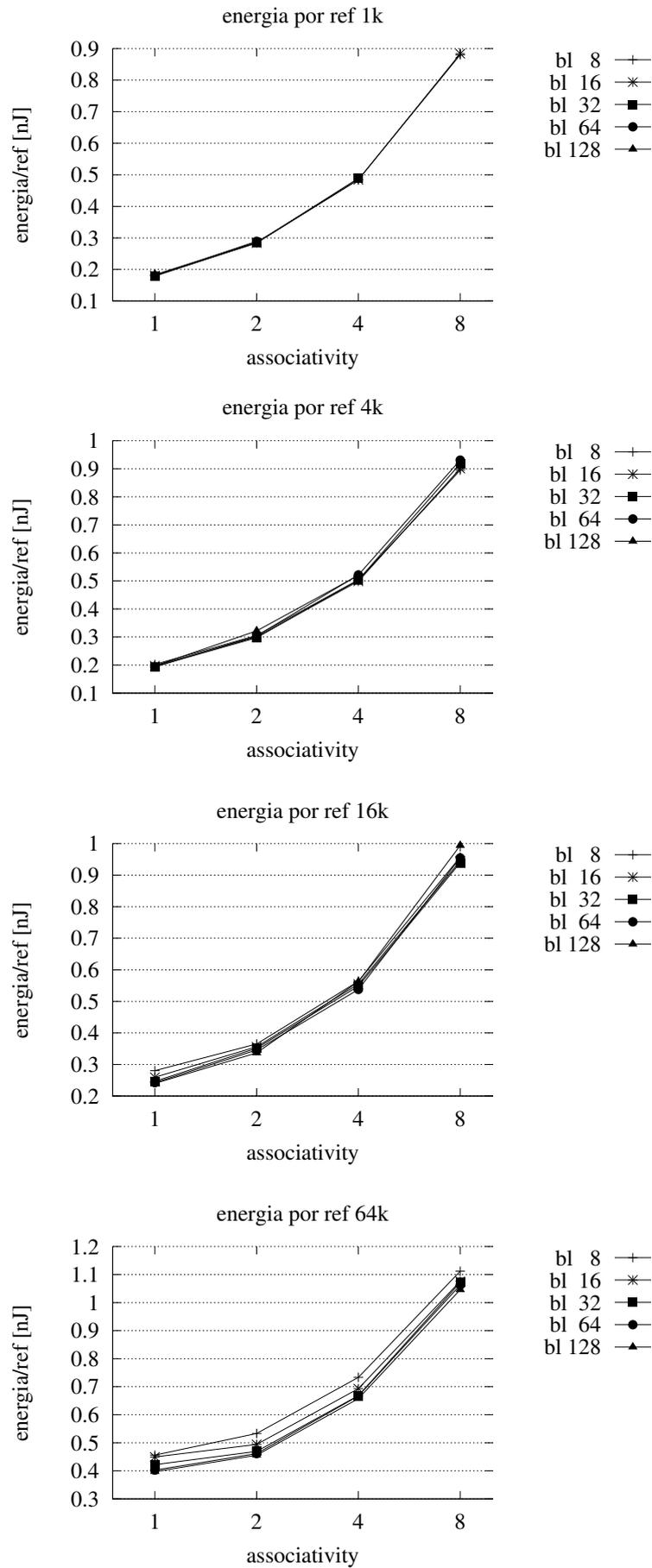


Figura 4: Energia/referência X capacidade X associatividade X tamanho de bloco.