

UMA FERRAMENTA COMPUTACIONAL PARA O PROJETO DE FORNOS DE PROCESSAMENTO TÉRMICO RÁPIDO

Edgar Alexander – edalex@iprj.uerj.br

Antônio José da Silva Neto * – ajsneto@iprj.uerj.br

Instituto Politécnico, Universidade do Estado do Rio de Janeiro, CP 97282, CEP 28601-970, Nova Friburgo, RJ, Brasil.

***Resumo.** Neste trabalho é apresentado um simulador de fornos de processamento térmico rápido desenvolvido em ambiente orientado a objetos onde através de uma interface amigável o usuário define a geometria do equipamento, realiza as simulações computacionais e visualiza os resultados obtidos. Desta forma o projetista dispõe de uma ferramenta computacional que permite testar várias configurações visando atender as condições de projeto. São apresentados, ainda, os resultados para uma configuração de forno para processamento térmico rápido, sendo feitas comparações com resultados obtidos através de outros simuladores. Também são discutidos alguns aspectos relativos ao processamento paralelo empregado no módulo de solução do problema de transferência de calor por radiação.*

***Palavras – chave :** Processamento térmico rápido, circuitos integrados, simulador computacional, condução de calor, radiação térmica.*

1. INTRODUÇÃO

A contínua evolução no processamento de semicondutores para a fabricação de circuitos integrados tem exigido o desenvolvimento de novos equipamentos. O uso de processamento térmico rápido (RTP – Rapid Thermal Processing) com radiação infravermelha tem sido objeto de intensa pesquisa. O Processamento Térmico Rápido – (Rapid Thermal Processing – RTP) se baseia na transferência de energia por radiação térmica para um objeto (Rozeboom, 1993). Uma variedade de fontes de calor tem sido usada em sistemas de RTP, sendo as lâmpadas de tungstênio-halogênio a mais utilizada. O objeto a ser aquecido dentro do forno de RTP, no processo de fabricação de circuitos integrados é um disco de silício com espessura variando entre meio e um milímetro e diâmetro padrão estando atualmente entre 6” e 8”, sobre o qual são depositados os filmes de produtos químicos.

Alguns autores, como Yu (1994), Kelly (1998) e Silva Neto et al. (1998) têm desenvolvido simuladores computacionais com o objetivo de testar determinadas configurações, tais como fornos com duas e três zonas de aquecimento. Tais simuladores, por

* Autor para correspondência.

ter o objetivo específico de simular uma determinada configuração, foram desenvolvidos sem a preocupação de serem utilizados por usuários distintos e em simulações de fornos distintos.

No presente trabalho é feita a descrição de um simulador computacional de forno de processamento térmico rápido, multiplataforma, através do qual é possível de forma amigável e através de interface gráfica, com utilização de recursos visuais, efetuar um teste simulado de configurações de fornos com aquecimento por radiação térmica. Com este simulador é possível avaliar na fase de projeto o desempenho de novas configurações para fornos de RTP que possibilitem uma temperatura uniforme no disco de silício. O simulador foi desenvolvido com ferramentas que lhe conferem um alto grau de portabilidade, podendo ser usado em máquinas com diferentes sistemas operacionais. Do ponto de vista de modelagem dos fenômenos envolvidos ele tem por base dois módulos centrais. No primeiro é calculada a transferência de calor por radiação térmica no interior do forno empregando um algoritmo de traçado de raios. Este algoritmo foi escolhido devido às características especulares dos refletores posicionados ao redor das lâmpadas de aquecimento. Após o cálculo do fluxo de calor que chega ao disco de silício é calculada a difusão de calor no mesmo com a solução do problema transiente de condução térmica. Neste segundo módulo é calculada a distribuição de temperaturas no disco de silício usando uma aproximação por diferenças finitas com formulação explícita.

2. MODELO TÉRMICO

A metodologia para simulação de fornos de RTP consiste basicamente de dois módulos. Quais sejam: (1) o cálculo do fluxo de calor na superfície e borda do disco de silício devido à fonte de calor radiante e (2) cálculo da distribuição de temperatura do disco dado o fluxo de calor calculado no módulo (1) (Alexander e Silva Neto, 1998). Outros módulos devem ser adicionados caso se deseje a análise da deposição de vapor químico ou oxidação térmica. No presente trabalho são tratados apenas os módulos (1) e (2).

A forma mais fácil e menos precisa para o módulo (1) é assumir irradiação uniforme na superfície do disco (Shieh e Carter, 1989). Devido à simplicidade da formulação, esta aproximação tem sido utilizada na investigação das interações ocorridas no forno e no sistema de controle (Sorrell et al., 1990).

Transferência de Calor por Radiação. Dentre as técnicas utilizadas para a modelagem da transferência de radiação térmica em cavidades, duas têm se destacado, uma utiliza o algoritmo de traçado de raios (Ray Trace) para simular o percurso percorrido pela radiação emitida pela lâmpada nas diversas direções para determinar a quantidade de radiação que chega ao disco de silício e a segunda se utiliza de métodos de Monte Carlo. Na realidade os dois algoritmos são bastante semelhantes sendo que no segundo são usados números randômicos para a tomada de decisões quanto aos eventos que ocorrem no interior do forno.

Em nosso simulador computacional utilizamos-nos do algoritmo de traçado de raios o qual é aplicado após a divisão do filamento de cada uma das lâmpadas de cada banco de lâmpadas em diversos segmentos. O algoritmo de traçado de raios é utilizado para simular o percurso que a radiação percorre após sair de cada um desses segmentos, em todas as direções. Este algoritmo é bastante difundido na área de computação gráfica quando da simulação de modelos de iluminação (Foley, 1996). Ao utilizar o traçado de raios em um modelo genérico, o algoritmo se preocupa com os raios refletidos e transmitidos. Em nosso modelo nos preocupamos apenas com os raios refletidos, já que não simulamos objetos semitransparentes. Deste modo, cada raio ao sair da lâmpada vai verificar se houve interseção com algum objeto do forno e ao interceptar um objeto, será determinado o seu ponto de interseção. Este ponto de interseção vai gerar um novo raio refletido e assim sucessivamente até que se chegue a um número de reflexões previamente estabelecido, quando o raio pára. A cada nova reflexão, o raio tem a sua potência reduzida, que pode ser maior ou menor dependendo do material de que o

objeto que foi interceptado é composto. Quando o objeto interceptado é o disco de silício, além da redução na potência do raio, adiciona-se a sua contribuição para o nó da malha computacional interceptada. Para o desenvolvimento de nosso algoritmo de traçado de raios utilizamos um algoritmo recursivo, ou seja, a função responsável pelo traçado de raios chama a si mesma dentro do seu próprio bloco de comandos, o que fornece uma solução elegante para o problema. Na Figura 1 é apresentada uma versão simplificada do algoritmo implementado computacionalmente. O raio lançado carrega consigo as informações sobre a sua origem, ou seja, de que banco de lâmpadas foi lançado. Deste modo possibilita que após o cálculo sejam feitos ajustes finos alterando a potência emitida de cada banco de lâmpadas verificando como estes ajustes afetam no aquecimento do disco.

```

Raytrace (Ent: reflexao, pos_inicial, direcao_raio, lampada_origem, potencia)
inicio
  se (reflexao <= qtdereflecoes) então
    incrementa reflexao
    Determina_Intersecao(Ent: pos_inicial, direcao_raio; Sai: objeto, num_obj, ponto_intersecao)
    se ocorreu interseção então
      nova_dir = Determina_Nova_Direção (Ent: pos_inicial,objeto,num_obj,ponto_intersecao);
      se (objeto = disco de silício) então
        calcula contribuição para aquecimento do disco
      fim-se
      nova_potencia = Calcula_Nova_Potencia(Ent: objeto, num_obj, potencia);
      Raytrace (Ent: reflexao, ponto_intersecao, nova_dir, lampada_origem, nova_potencia);
    fim-se
  fim-se
fim-raytrace

```

Figura 1 - Algoritmo recursivo de traçado de raios

Transferência de Calor por Condução. Após ter sido calculada a radiação que chega ao disco de silício oriunda das fontes térmicas, ou seja, dos bancos de lâmpadas, devemos calcular a distribuição da temperatura no disco de silício. A distribuição da temperatura transiente sobre o disco é obtida através da solução da equação de difusão de calor com condição inicial e de contorno apropriadas. A equação de condução de calor é escrita em coordenadas cilíndricas como

$$\frac{1}{r} \frac{\partial}{\partial r} (k r \frac{\partial T}{\partial r}) + \frac{1}{r^2} \frac{\partial}{\partial \phi} (k \frac{\partial T}{\partial \phi}) + g(r, \phi, t) = \rho c_p \frac{\partial T}{\partial t} \quad (1)$$

com

$$g(r, \phi, t) = \frac{1}{l} [\varepsilon (q''_r - \sigma T^4) - q''_c] \quad (2)$$

onde $T(r, \phi, t)$ é a temperatura do disco, $\rho(T)$ é a densidade do silício, $k(T)$ é a condutividade térmica, $c_p(T)$ é o calor específico do silício, $q''_r(r, \phi, t)$ é o fluxo de calor, irradiação total, que chega ao disco, podendo vir diretamente dos filamentos das lâmpadas ou estar chegando a partir de uma reflexão especular na parede do forno, $q''_c(r, \phi, t)$ é o fluxo de calor perdido por convecção, ε é a emissividade do disco e σ é a constante de Stefan-Boltzmann.

A irradiação total $q''_r(r, \phi, t)$ foi calculada com o algoritmo de traçado de raios descrito anteriormente, o qual levou em consideração múltiplas reflexões especulares ocorridas no interior do forno. Na realidade o termo $\varepsilon q''_r$ é obtido no início do módulo de cálculo da distribuição de temperaturas no disco de silício, a partir da totalização da potência absorvida por cada célula da malha computacional dividida pela área da célula.

Note que as condições de contorno de radiação e convecção são incluídas no termo fonte equivalente $g(r, \phi, t)$. Um termo fonte similar é derivado para os elementos diferenciais da borda do disco. Note também que na dedução da Eq.(1) foi assumida a temperatura uniforme através da espessura do disco, ou seja não há variação com o eixo z . Isto é justificado pelo raio do disco ser bem maior que a espessura do mesmo.

A temperatura inicial do disco de silício é conhecida.

$$T(r, \phi, t) = T_i \quad \text{para todo } r, \phi \text{ e } t = 0 \quad (3)$$

De modo a obter uma solução aproximada das Eqs.(1)-(3), uma discretização por diferenças finitas com formulação explícita é utilizada. As propriedades térmicas não apresentam variações abruptas com a temperatura, e o tempo total de processamento rápido é curto, ou seja, inferior a um minuto. Deste modo, o passo no tempo a ser usado para assegurar estabilidade não necessita ser extremamente pequeno, não apresentando, portanto, problemas em relação ao tempo computacional, justificando o uso da formulação explícita.

Para os nós interiores da malha escreve-se

$$\begin{aligned} T_{i,j}^{n+1} = & T_{i,j}^n + \left(\frac{\Delta t}{\rho_{i,j}^n c_{p,i,j}^n} \right) \left(\frac{1}{r_i \Delta r^2} \right) [k_{i+\frac{1}{2},j}^n r_{i+\frac{1}{2}} (T_{i+1,j}^n - T_{i,j}^n) - k_{i-\frac{1}{2},j}^n r_{i-\frac{1}{2}} (T_{i,j}^n - T_{i-1,j}^n)] \\ & + \left(\frac{\Delta t}{\rho_{i,j}^n c_{p,i,j}^n} \right) \left(\frac{1}{r_i^2 \Delta \phi^2} \right) [k_{i,j+\frac{1}{2}}^n (T_{i,j+1}^n - T_{i,j}^n) - k_{i,j-\frac{1}{2}}^n (T_{i,j}^n - T_{i,j-1}^n)] + \left(\frac{\Delta t}{\rho_{i,j}^n c_{p,i,j}^n} \right) g_{i,j}^n \end{aligned} \quad (4)$$

$2 \leq i \leq M-1, \quad 2 \leq j \leq N-1$

onde

$$r_i = (i-1) \Delta r, \quad i = 1, 2, \dots, M \quad (5)$$

$$\phi_j = (j-1) \Delta \phi, \quad j = 1, 2, \dots, N \quad (6)$$

$$t_n = n \Delta t, \quad n = 1, 2, \dots, L \quad (7)$$

$$\Delta r = \frac{R}{M-1} \quad (8)$$

$$\Delta \phi = \frac{2\pi}{N} \quad (9)$$

$$k_{i+\frac{1}{2},j}^n = \frac{1}{2} (k_{i,j}^n + k_{i+1,j}^n) \quad (10)$$

onde R é o raio do disco e os outros três valores utilizados para a condutividade térmica na equação (5), $k_{i-\frac{1}{2},j}^n$, $k_{i,j+\frac{1}{2}}^n$ e $k_{i,j-\frac{1}{2}}^n$, são obtidos de forma similar a $k_{i+\frac{1}{2},j}^n$ mostrada na equação (10). A equação (4) não pode ser utilizada para o cálculo no nó central por que r vai a zero. Deste modo, consideramos uma média dos nós vizinhos, ficando da seguinte forma (Özisik, 1993)

$$T_c^{n+1} = T_c^n + \left(\frac{\Delta t}{2 \rho_c^n c_{p,c}^n \Delta r^2} \right) [4 \bar{T}_j^n k_c^n - T_c^n (4 k_c^n + 4 \bar{k}_j^n) + 4 \overline{k_j T_j^n}] + \left(\frac{\Delta t}{\rho_c^n c_{p,c}^n} \right) g_c^n \quad (11)$$

onde \bar{T}_j^n , \bar{k}_j^n e $\overline{k_j T_j^n}$ representam as médias da condutividade térmica, da temperatura e do produto da condutividade térmica e temperatura, respectivamente, para os nós em volta do centro do disco.

Na malha de diferenças finitas ainda são incluídos nós fictícios de raio $r = M \Delta r$, além da borda do disco. A temperatura desses nós é obtida pela condição de fronteira na borda. Uma vez conhecidas estas temperaturas, T_{M+1}^n é conhecido e a Eq. (4) é escrita para o nó $i = M$. Fica

então um conjunto completo de equações para se determinar as temperaturas $T_{i,j}^{n+1}$ com $i = 1, 2, \dots, M$ e $j = 1, 2, \dots, N$, para $n = 1, 2, \dots, L$.

Para $i = 1$,

$$T_{1,j}^{n+1} = T_c^{n+1}, \quad j = 1, 2, \dots, N. \quad (12)$$

3. SIMULADOR COMPUTACIONAL DESENVOLVIDO PARA FORNOS DE RTP

O simulador computacional desenvolvido e denominado RaThePS (Rapid Thermal Processing Simulator), consiste basicamente de interfaces gráficas para entrada de dados e visualização gráfica do forno e dos resultados numéricos, e das rotinas computacionais para o cálculo da transferência radiativa e da distribuição de temperaturas no disco de silício, conforme os modelos descritos na seção 2.

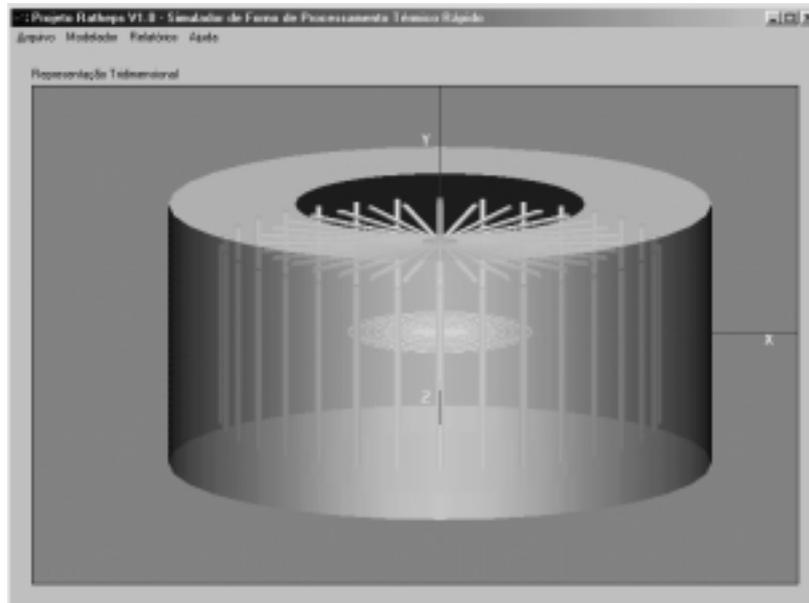


Figura 2 - Tela Principal do Simulador de RTP

As saídas do sistema. Após o cálculo da transferência de calor entre a fonte térmica e o disco de silício e após a distribuição de temperatura no disco, os resultados são apresentados em diversos relatórios diferentes. Como relatórios temos: (i) um mapa dinâmico, mostrando a distribuição do fluxo de calor absorvido no disco e que permite que sejam feitos ajustes na intensidade da radiação de cada banco de lâmpadas de modo a conseguir melhorar a uniformidade na distribuição; (ii) um gráfico dinâmico o qual também permite o ajuste da intensidade e permite a visualização da distribuição em um gráfico em forma de gangorra onde o ponto central se mantém fixo e os demais pontos sobem ou descem conforme maior ou menor for a quantidade de radiação recebida; (iii) um relatório com a configuração do forno; e (iv) um relatório com os resultados numéricos da distribuição do fluxo de calor absorvido no disco de silício.

Mapa dinâmico. O mapa dinâmico, conforme mostrado na Figura 3, representa através de cores variações na quantidade de radiação (potência) recebida pelo disco de silício. Ao lado existem controles que permitem ajustar a intensidade de cada banco de lâmpadas que pode variar de 100% a 0%. Abaixo e à direita existe uma janela onde é informada a quantidade de radiação recebida por anel, vinda de cada banco de lâmpadas e o seu total. Estes valores se modificam conforme se aumenta ou diminui a intensidade de cada banco de lâmpadas.

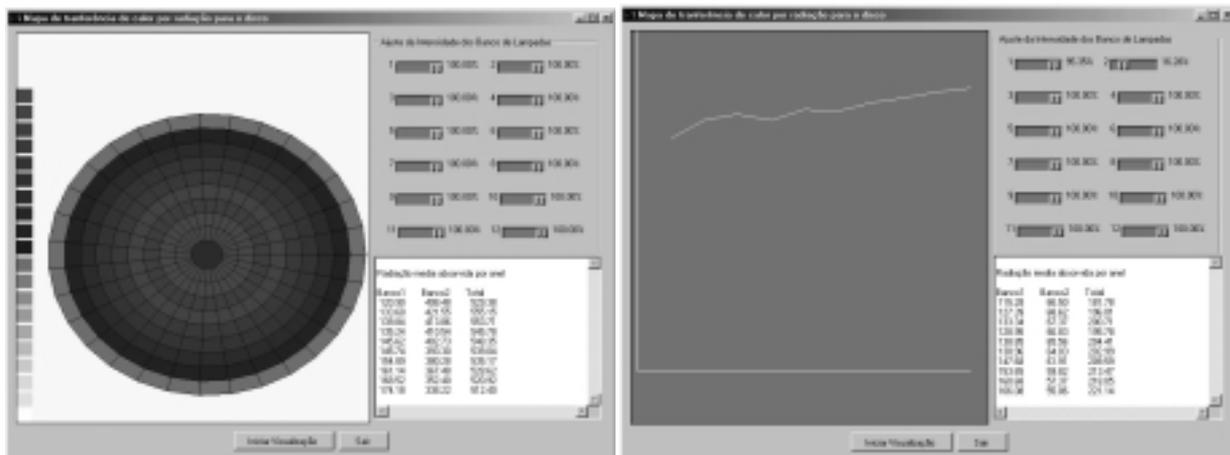


Figura 3 - Mapa e Gráfico Dinâmico de Transferência de Calor por Radiação para o Disco de Silício

Gráfico Dinâmico. O gráfico dinâmico de transferência de calor, mostrado na Figura 3, permite visualizar a quantidade de radiação (potência) recebida pelo disco, sendo que o lado esquerdo do gráfico representa a parte central do disco de silício e o lado direito a borda. Deste modo, a parte esquerda do gráfico mais baixa do que a parte direita, por exemplo, indica que o disco está mais aquecido na borda do que em seu centro. Assim como no Mapa existem controles situados no lado direito e que permitem ajustar a intensidade de cada banco de lâmpadas. O resultado numérico é mostrado em uma janela abaixo e à direita na qual a intensidade da radiação recebida pelo disco aparece separada por banco de lâmpadas e totalizada. Após o ajuste, alterando a intensidade emitida por cada banco de lâmpadas pode-se voltar ao módulo onde é efetuado o cálculo da distribuição de temperaturas e executá-lo, verificando a contribuição do ajuste da intensidade dos bancos de lâmpadas na uniformidade da distribuição de temperaturas no disco de silício.

Outros Relatórios. Além do gráfico e do mapa dinâmico acima descritos, o sistema fornece ainda as saídas numéricas com informações sobre a transferência de calor ocorrida entre a fonte radiativa e o disco de silício.

Linguagem. A linguagem escolhida para implementar o simulador computacional de fornos de processamento térmico rápido foi o C++. Diversos fatores levaram à escolha dessa linguagem. Primeiramente é uma linguagem amplamente difundida, padronizada, moderna e utilizada no desenvolvimento de grande parte dos softwares comerciais. Depois, por ser uma linguagem orientada a objetos, permite uma melhor simulação dos objetos do mundo real (Stroustrup, 1997). Por ser uma linguagem compilada e próxima do sistema operacional é uma linguagem que permite maior velocidade no processamento. Outro fator que foi considerado na escolha da linguagem foi a existência de bibliotecas de funções que permitiriam desenvolver uma interface gráfica portátil, de modo a facilitar migrar o programa entre equipamentos com sistemas operacionais diversos, e a possibilidade de paralelizar o programa, fazendo com que o mesmo pudesse ser executado em diversos processadores diferentes de modo a aumentar a velocidade de processamento.

Sistema de interface com o usuário IUP. O IUP, desenvolvido no Laboratório Tecgraf do Departamento de Informática da PUC-RJ, é um sistema portátil de interface com usuário composto por uma Linguagem de Especificação de Diálogos (LED) e uma biblioteca de aproximadamente 40 funções para a criação e a manipulação de diálogos (Levy, 1993).

A proposta do IUP é permitir que um programa possa ser executado sem modificações em qualquer sistema de interface, conferindo-lhe uma alta portabilidade. Os sistemas operacionais utilizados são os mais distintos possíveis tais como: DOS, X-Windows,

OpenLook, Motif, Microsoft Windows e Macintosh, possibilitando que um sistema seja desenvolvido para rodar tanto em um microcomputador PC com DOS, Windows ou Linux, quanto em uma estação de trabalho tendo como sistema operacional SunOS (Sun), IRIX (Silicon Graphics), ULTRIX (Digital) ou ainda AIX (IBM).

O IUP foi utilizado no desenvolvimento do simulador RaThePS de modo a permitir a criação de uma interface de usuário que fosse a mais independente possível do hardware e do sistema operacional e que permitisse migrar o programa de uma plataforma como o PC utilizando o sistema operacional Windows 98, que foi a plataforma utilizada para o desenvolvimento do programa, para outra, como a Silicon Graphics utilizando IRIX como sistema operacional, que foi a plataforma utilizada para rodar o programa simultaneamente em vários processadores.

OpenGL. Para a criação da representação tridimensional e para as demais saídas gráficas, tais como os gráficos e os mapas dinâmicos foi utilizado o OpenGL. O OpenGL foi criado nos laboratórios da Silicon Graphics e consiste de uma biblioteca para aplicações gráficas com aproximadamente 150 comandos distintos que devem ser utilizados para especificar os objetos e operações necessárias para produzir aplicações interativas tridimensionais (Woo et al., 1997). OpenGL foi desenvolvido como uma interface independente do *hardware* de modo a poder ser implementado em várias plataformas de *hardware* diferentes. Para alcançar esta qualidade nenhum comando para execução e gerenciamento de janelas ou para obtenção de dados pelo usuário foram incluídos no OpenGL. De forma análoga, OpenGL também não fornece comandos de alto nível para descrever modelos de objetos tridimensionais. Com OpenGL os modelos devem ser desenhados a partir de um pequeno conjunto de primitivas geométricas – pontos, linhas e polígonos.

4. PROCESSAMENTO PARALELO

Face ao elevado custo computacional do algoritmo de traçado de raios empregado no módulo de solução do problema de transferência de calor por radiação, optou-se por fazer uma versão do software utilizando processamento paralelo no cálculo desta rotina.

No processamento paralelo é utilizado um conjunto de processadores capaz de trabalhar cooperativamente na solução de um problema computacional. Esta definição abrange desde supercomputadores com centenas ou milhares de processadores, redes de estações de trabalho, estações de trabalho multiprocessadoras até sistemas acoplados ou interligados tipo hipercubo. (Toledo e Silva, 1997). Existem diversos softwares, tais como o MPL, o MPI e o PVM, criados para configurar um grupo heterogêneo de computadores interconectados por uma ou mais redes como uma máquina virtual (MacDonald et al., 1992). O software utilizado no simulador computacional apresentado neste trabalho para implementar o processamento paralelo foi o MPI. Sua escolha se deu face ao mesmo estar se tornando padrão no uso de processamento paralelo com estações de trabalho e ser multiplataforma.

Análise de Algoritmos Paralelos. Para avaliar o desempenho de um programa paralelo deve-se considerar diversos parâmetros tais como: tempo de execução, eficiência e *speedup*.

A eficiência do programa pode ser definida como a fração do tempo em que os processadores passam realizando trabalho útil. Deste modo a eficiência nos fornece, em alguns casos, uma forma mais conveniente de medir a qualidade de algoritmos paralelos. O “Speedup” está relacionado com a eficiência do programa paralelo e representa o quanto foi reduzido o tempo de execução através da utilização de um certo número de processadores (Sardiña et al., 1999). O “Speedup” (S_u) é obtido da seguinte forma

$$S_u = \frac{t_s}{t_p} \quad (1)$$

onde t_s , é o tempo de CPU para o cálculo feito com apenas um processador e t_p é o tempo de

CPU para o cálculo feito com p processadores. A eficiência (η) no uso de p processadores é definida como

$$\eta = \frac{S_u}{p} \quad (2)$$

Em uma situação ideal teríamos,

$$t_s = p t_p \longrightarrow S_u = p \longrightarrow \eta = 1 \quad (3)$$

4.1. Resultados com Processamento Paralelo

Tabela 1 - Resultados com Processamento Paralelo

Processadores	Tempo de CPU (s)	S_u	η
1	96	-	-
2	48	2,00	1,00
4	25	3,84	0,96

Estes resultados foram obtidos simulando uma configuração simplificada, tendo apenas um banco de lâmpadas, 300 divisões polares para cada segmento e 4 níveis de reflexão admitidas para cada raio lançado. Em algumas configurações mais complexas, como a simulação de um forno de 3 zonas de aquecimento (Silva Neto et al., 1998) com 3000 divisões polares para cada segmento e 4 níveis de reflexão, o tempo computacional para o cálculo da transferência de calor por radiação térmica com o simulador RaThePS foi superior a 10 horas. Nestes casos a eficiência do programa rodando em paralelo seria ainda melhor.

5. ALGUNS RESULTADOS OBTIDOS COM O SIMULADOR COMPUTACIONAL

Foram realizados vários testes com o RaThePS procurando simular experimentos feitos com simuladores desenvolvidos em outras instituições de pesquisa, além de simulações comparativas com dados experimentais reais. Os testes se mostraram satisfatórios. Em um desses testes foi simulado um experimento feito com um simulador de forno de RTP desenvolvido na Universidade do Estado da Carolina do Norte (Fordham, 1997), simulando um forno de processamento térmico rápido com uma zona de aquecimento, localizada acima do disco de silício. Neste teste foram feitas diversas simulações variando a densidade dos raios lançados, ou seja, variou-se a quantidade de lançamento de raios de cada segmento do filamento da lâmpada, de 50 ângulos polares até 5000 ângulos polares. Nas Figuras 4 e 5 vemos o resultado obtido na Universidade do Estado da Carolina do Norte e o resultado obtido através do RaThePS respectivamente. O teste mostrou que ambos os simuladores tiveram resultados bastante semelhantes a partir de 300 ângulos polares.

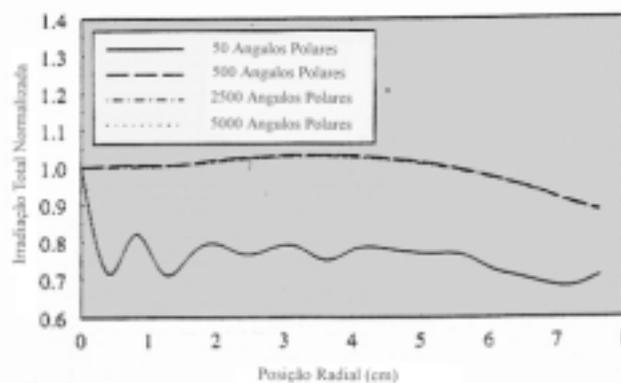


Figura 4 - Simulação realizada na Universidade do Estado da Carolina do Norte

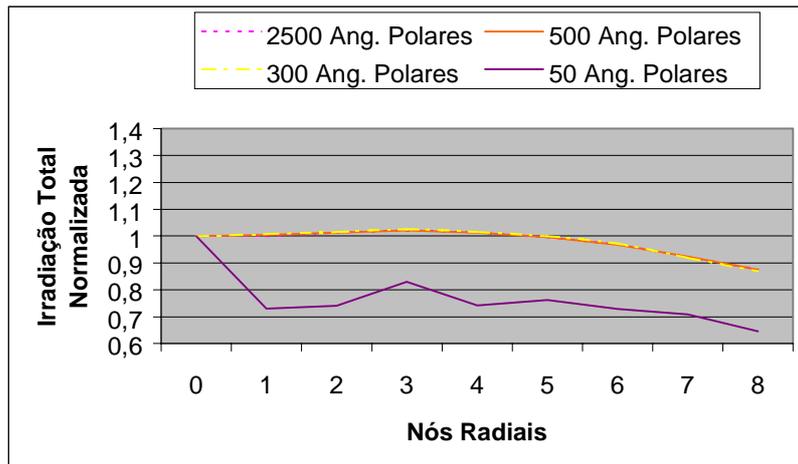


Figura 5 - Simulação realizada com o RaThePS

6. CONCLUSÃO

O simulador de fornos de RTP com interfaces e saídas gráficas RaThePS mostrou-se uma ferramenta útil ao apresentar resultados semelhantes ao de outros simuladores e de dados experimentais reais. O desenvolvimento das saídas gráficas permitiu uma maior rapidez na avaliação dos resultados e a sua interface, com a visualização tridimensional do forno, possibilita a eliminação de erros na entrada de dados.

A utilização de bibliotecas computacionais portáteis e linguagem padronizada permitiu que o sistema fosse implementado em sistemas operacionais e equipamentos distintos sem que fossem necessárias grandes alterações nos códigos dos programas.

O simulador desenvolvido mostrou-se bastante genérico, de modo a possibilitar a simulação de diversas configurações diferentes de fornos de RTP. Sua restrição se deve apenas à não implementação de algumas formas geométricas, tais como refletores cônicos ou retangulares, os quais aumentariam ainda mais a possibilidade de configurações a serem simuladas.

Apesar de estar apresentando bons resultados, o desempenho do simulador ainda é baixa, quando simulados fornos em plataformas PC convencionais, em máquinas mono processadas, com os atuais processadores com clock de até 400Mhz.

Para diminuir as dificuldades associadas ao baixo desempenho computacional do algoritmo de traçado de raios, o uso do processamento paralelo se mostrou uma opção viável, face ao algoritmo ser totalmente paralelizável, não necessitando, portanto de alta taxa de comunicação entre os processadores. Deste modo é possível distribuir o algoritmo entre um grande número de máquinas chegando a conseguir uma situação bem próxima do ideal, ou seja, eficiências próximas a 100%. Conforme visto nos resultados apresentados em algumas simulações o tempo computacional não é excessivo, mas em alguns casos com elevado número de bancos de lâmpadas e de divisões no domínio do ângulo polar, este pode se tornar proibitivo.

REFERÊNCIAS

Alexander, E. e Silva Neto, A. J., 1998, A Modelagem de Fornos de Processamento Térmico Rápido Usados na Fabricação de Circuitos Integrados, Anais do I Encontro de Modelagem Computacional, Uiversidade do Estado do Rio de Janeiro, Campus Regional de Nova Friburgo, Instituto Politécnico.

- Foley, J. D., 1996, Computer Graphics Principles and Practice, Addison-Wesley Publishing Company.
- Fordham, J. M., 1997, Thermal Modeling and Model Validation for Semiconductor Wafer Rapid Thermal Processor Design, Tese de Doutorado, North Carolina State University.
- Kelly, S. J., 1998, Design Methodology and Design of the Lamps and Reflectors for a Two-Zone Rapid Thermal Processor, Dissertação de Mestrado, North Carolina State University.
- Levy, C. H., 1993, IUP/LED: Uma Ferramenta Portátil de Interface com Usuário, Dissertação de Mestrado, Departamento de Informática, PUC-Rio.
- MacDonald, N. Minty, E., Harding, T. e Brown, S., 1992, Writing Message-Passing parallel Programs with MPI, Edinburgh Parallel Computing Centre, The University of Edinburgh.
- Özisik, M. N., 1993, Heat Conduction, John Wiley & Sons, New York.
- Roozeboom F., 1993, Manufacturing Equipment Issues in Rapid Thermal Processing. In Rapid Thermal Processing Science and Technology. Academic Press, Inc.
- Sardiña, I. M., Henderson, L. N., Silva Neto, A. J., 1999, Solução da Equação de Condução de Calor por Decomposição de Domínio, XV Congresso Brasileiro de Engenharia Mecânica, Águas de Lindóia, São Paulo.
- Shieh, T. J. e Carter, R. L., 1989, RAPS – A Rapid Thermal Processor Simulation Program, IEEE Transactions on Electron Devices, Vol. 36, No. 1, pp. 19-24.
- Silva Neto, A. J., Fordham, M. J., Kiether, W. J. e Sorrell, F. Y., 1998, Rapid Thermal Processing Furnace with Three Heating Zones, North Carolina State University, In Journal of the Brazilian Society of Mechanical Sciences, Vol. XX, No. 4, pp.532-541.
- Sorrell, F. Y., Harris, A. e Gyurcsik, R. S., 1990, A Global Model for Rapid Thermal Processors, IEEE Transactions on Semiconductor Manufacturing, Vol. 3, No. 4, pp. 183-188.
- Stroustrup, B., 1997, The C++ Programming Language, Addison-Wesley Longman Inc.
- Toledo, E. M. e Silva, R. S., 1997, Introdução à Computação Paralela, Laboratório Nacional de Computação Científica.
- Woo, M., Neider, J., Davis, T., 1997, OpenGL Programming Guide: The Official Guide to Learning OpenGL, Addison-Wesley Developers Press.
- Yu, S., 1994, Rapid Thermal Processor Simulation and Optimization, Tese de Doutorado, North Carolina State University.

Abstract. *In this work is presented a rapid thermal processing furnace simulator developed with object oriented programming, where the user can define the geometry from the furnace, perform the simulation and watch the results through user friendly graphical interfaces. In this way the designer have a computational tool that allows the testing of several furnaces configurations. A comparison of the results obtained with other simulators is presented for a rapid thermal processing furnace configuration. We also examine some aspects of parallel processing adopted in the module of radiative heat transfer computation.*

Key-words: Rapid Thermal Processing, Integrated circuit, Computational simulator, Conduction heat transfer, Radiative heat transfer.