

## 5 Resultados Experimentais

Neste capítulo são apresentados alguns testes realizados tendo em vista julgar a aplicação desenvolvida em algumas das situações em que ela possa vir a ser utilizada, assim como verificar e justificar a adoção dos algoritmos propostos.

Inicialmente apresenta-se uma descrição das condições nas quais os testes foram realizados. Em seguida são apresentados os resultados obtidos da avaliação de desempenho de visualização da aplicação. Por fim, os resultados de desempenho do gerenciamento de ladrilhos são apresentados.

### 5.1 Descrição dos Testes

Com o objetivo de mostrar que o desempenho do algoritmo executa é satisfatório mesmo em máquinas de baixo poder computacional, todos os testes foram realizados utilizando um *notebook* com a configuração especificada na Tabela 5.1.

<b>CPU</b>	<b>Intel Pentium<sup>®</sup> 4 2.80 GHz com Hyperthread</b>
<b>Memória</b>	<b>512 DDR SDRAM</b>
<b>GPU</b>	<b>nVidia GeForce<sup>™</sup> Fx Go5200 com 64MB VRAM</b>
<b>Sistema Operacional</b>	<b>Microsoft Windows XP Media Center Edition</b>

Tabela 5.1: Configuração do computador dos testes

Todos os testes foram realizados utilizando uma janela de visualização na resolução  $512 \times 512$ . A visualização é sempre feita com uma tolerância de erro projetado igual a 1 pixel. O terreno utilizado é um dado real, representando uma região de Washington, nos Estados Unidos, chamada Puget Sound [23]. Foram utilizadas duas amostragens deste dado. A primeira tem

resolução  $2049 \times 2049$ , contendo aproximadamente 4,2 milhões de amostras, sendo usada somente no teste da Seção 5.2.2. Em todos os outros testes, é utilizado um dado com resolução  $16.385 \times 16.385$ , que contém aproximadamente 267,9 milhões de amostras. Nesta amostragem, a região representada apresenta um espaçamento de 10 metros entre cada amostra, possuindo portanto uma extensão de 163,385 quilômetros em cada direção. Em todos os testes utilizou-se também uma textura de resolução  $1024 \times 1024$  mapeada sobre a superfície do terreno.

Com o intuito de avaliar o algoritmo em diversas situações, foi traçado um caminho conforme mostrado na Figura 5.1. O caminho está definido com alguns pontos-chave, cada um dos quais determina uma posição, uma direção e um tempo. As posições intermediárias em um dado tempo são determinadas por uma *spline* que passa pelos pontos-chave. De forma análoga, é feita a interpolação das direções, utilizando quatérnios. Portanto, a navegação neste caminho em todos os testes é feita baseada em tempo.

O caminho está dividido em trechos que procuram analisar características diferentes do algoritmo. Buscou-se também apresentar um comportamento comum de navegação de um usuário do sistema. A Tabela 5.2 apresenta as características de cada trecho observado na Figura 5.1.

Trechos do caminho		
Intervalo	Descrição	Tempo
Entre (a) e (b)	Navegação rasante em região plana	0 a 17,9 (s)
Entre (b) e (c)	Giro	17,9 a 28,8 (s)
Entre (c) e (d)	Alta velocidade para região montanhosa	28,8 a 34,1 (s)
Entre (d) e (e)	Subida	34,1 a 41,1 (s)
Entre (e) e (f)	Mergulho rápido para próximo ao vulcão	41,1 a 45,7 (s)
Entre (f) e (g)	Giro em torno do vulcão	45,7 a 58,6 (s)
Entre (g) e (h)	Alta velocidade para região de vale	58,6 a 74,2 (s)
Entre (h) e (i)	Navegação em região de vale	74,2 a 86,0 (s)

Tabela 5.2: Trechos de navegação do caminho utilizado nos testes (são apresentados o tipo de navegação e o tempo de cada trecho)

## 5.2

### Testes de Desempenho da Visualização

Os testes desta seção têm como objetivo analisar o desempenho e verificar o correto funcionamento dos algoritmos de visualização apresentados no trabalho. São analisadas as taxas de quadros por segundo de terrenos iguais com ladrilhos de tamanhos diferentes, de um algoritmo conhecido con-

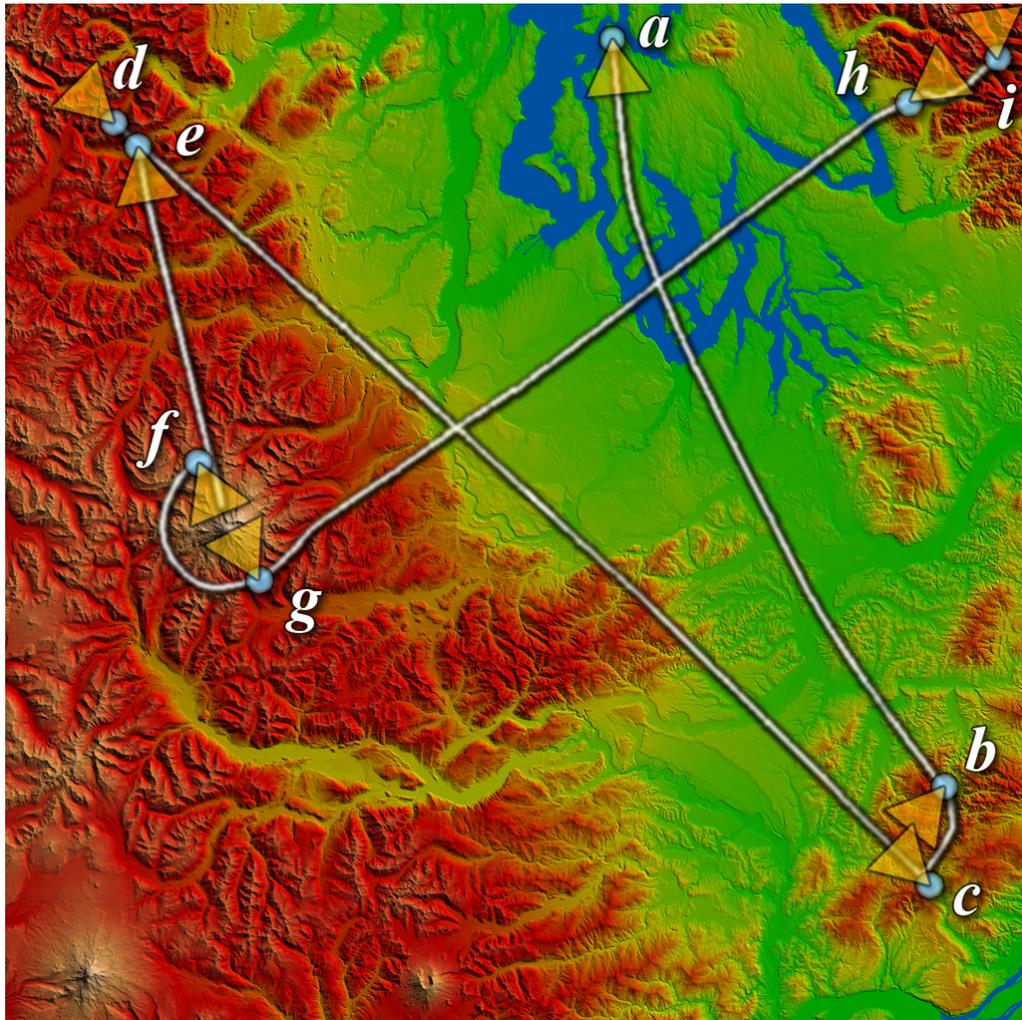


Figura 5.1: Visão aérea do terreno. Os pontos marcados mostram os trechos do caminho divididos de acordo com as características de navegação. A direção de visualização em cada ponto também está indicada. Uma descrição dos trechos pode ser encontrada na Tabela 5.2

tra o desenvolvido neste trabalho e do algoritmo que limita a quantidade de polígonos a ser usada.

### 5.2.1 Tamanho de Ladrilhos

Este teste consiste em verificar a diferença de desempenho da visualização em um terreno grande utilizando duas resoluções de ladrilhos diferentes,  $17 \times 17$  e  $33 \times 33$ . Quando utiliza-se uma resolução de ladrilhos muito pequena, a *quadtree* fica muito profunda, exigindo um esforço maior da *CPU* para o percurso, cálculo de erro, visibilidade, entre outros, além de aumentar o consumo de memória para o armazenamento dos metadados. Em contrapartida, o procedimento de refinamento, por apresentar uma

granularidade maior, produz uma malha com menos triângulos, aliviando a *GPU*. O objetivo deste teste é, portanto, comparar granularidades diferentes para a execução do algoritmo, procurando um balanceamento ideal entre o uso de *CPU*/memória e *GPU*.

Média da Taxa de Quadros por Segundo	
Ladrilhos de 33x33	73,26
Ladrilhos de 17x17	128,32

Tabela 5.3: Média ao longo do caminho da taxa de quadros por segundo

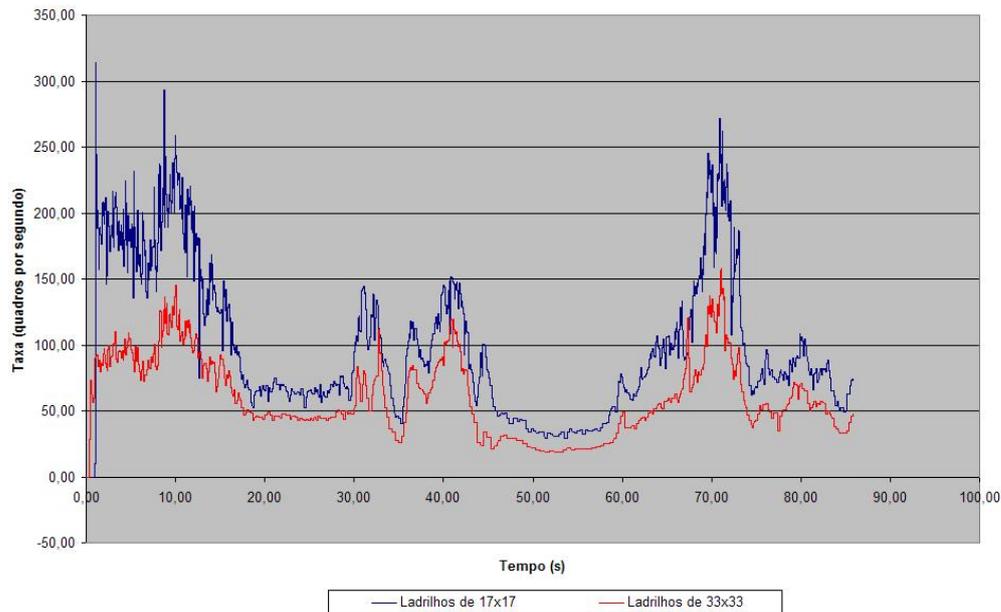


Figura 5.2: Taxa de quadros por segundo de terrenos com ladrilhos 17x17 e 33x33

Conforme ilustrado na Figura 5.2 e pelas médias apresentadas na Tabela 5.3, pode-se concluir que o uso de ladrilhos de dimensão 17x17 é o melhor no caso de teste apresentado. Entretanto, o resultado acima pode variar de acordo com o terreno usado, o poder computacional de *CPU* e *GPU* e a capacidade de memória.

## 5.2.2 Comparação com o Algoritmo de Lindstrom

O experimento desta seção faz uma comparação entre o algoritmo apresentado neste trabalho e uma implementação do algoritmo de Lindstrom *et al.* [18] desenvolvida no início desta pesquisa. O objetivo é analisar o desempenho do algoritmo deste trabalho em memória principal. Para isso,

o terreno utilizado tem resolução total de  $2049 \times 2049$ , com ladrilhos na resolução  $17 \times 17$ .

Média da Taxa de Quadros por Segundo	
<b>Algoritmo deste trabalho</b>	187,20
<b>Lindstrom</b>	324,70

Tabela 5.4: Média ao longo do caminho da taxa de quadros por segundo

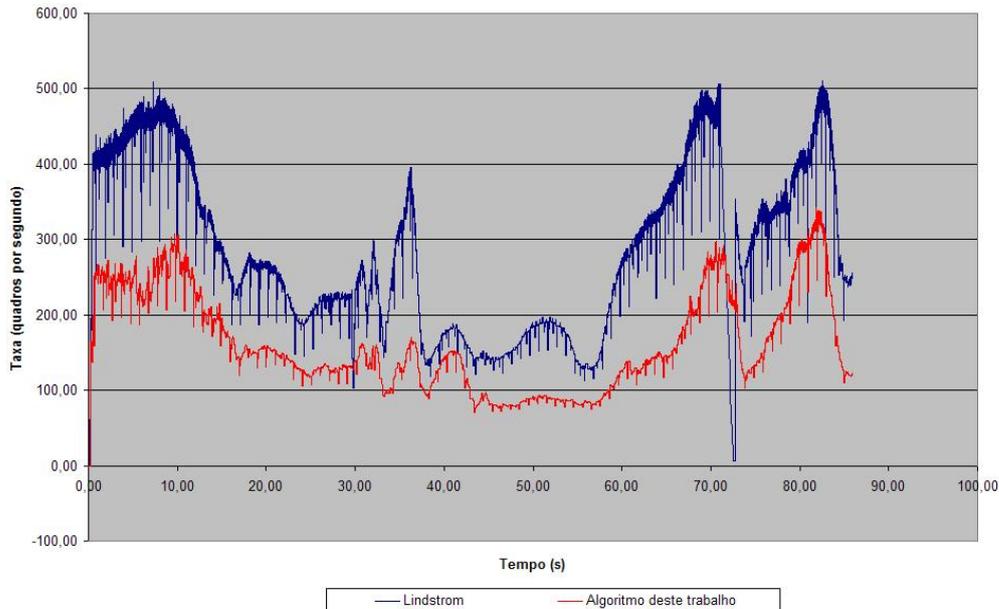


Figura 5.3: Taxa de quadros por segundo do algoritmo deste trabalho e de Lindstrom

Pela Figura 5.3 e pela Tabela 5.4 pode-se perceber que o algoritmo de Lindstrom superou o algoritmo apresentado neste trabalho. Isto já era esperado, pois o algoritmo de Lindstrom tem como prioridade gerar uma malha ótima, enquanto no algoritmo deste trabalho procura-se um uso menor de memória e *CPU*. No entanto, o algoritmo de Lindstrom não é escalável e o uso de um terreno maior para os testes não foi possível no computador utilizado.

### 5.2.3 Visualização Limitada por Quantidade de Polígonos

Este teste tem o objetivo de verificar a implementação do algoritmo apresentado na Seção 4.2.1. Como nos testes anteriores, o caminho é percorrido coletando a taxa de quadros por segundo. Foi especificada uma quantidade de polígonos máxima, de forma a garantir uma taxa mínima

de quadros por segundo (neste teste foi alcançada uma taxa mínima de 60 quadros por segundo). O gráfico abaixo compara a navegação com e sem a limitação imposta.

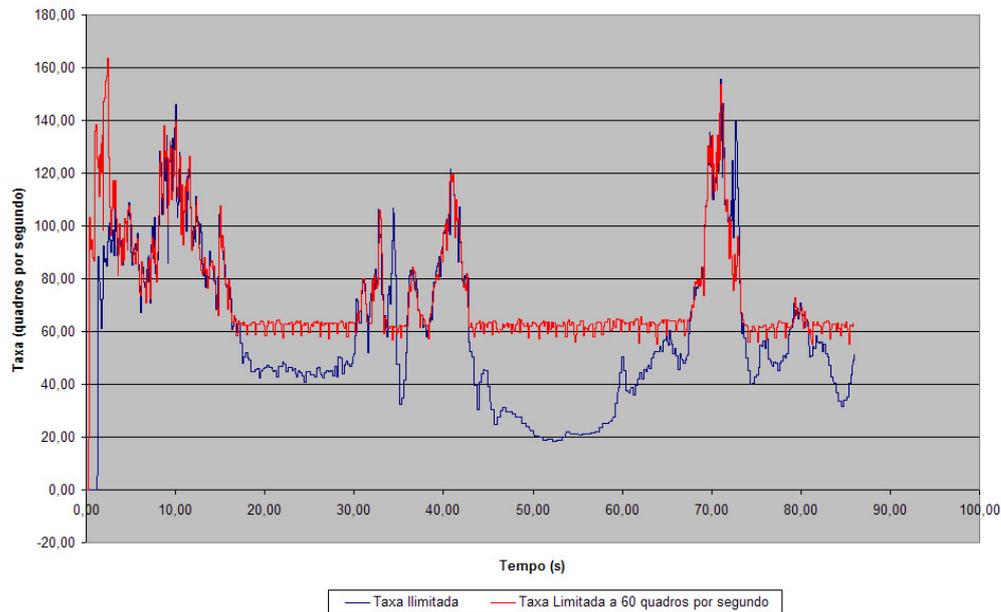


Figura 5.4: Taxa de quadros por segundo comparando o algoritmo normal com o algoritmo de limite de polígonos

Pode-se observar pelo gráfico que o algoritmo atingiu o objetivo a que se propôs, sempre mantendo a taxa mínima especificada. Vale ressaltar que a taxa pode ficar acima da especificada e a manutenção de uma taxa constante, se for o desejado, cabe à aplicação implementar.

### 5.3

#### Testes de Desempenho do Gerenciamento de Ladrilhos

Esta seção visa fazer uma comparação do desempenho da linha de execução de gerenciamento de ladrilhos com e sem o uso de predição. São feitas comparações utilizando erros médios e máximos, além da porcentagem de erro de ladrilhos não presentes em memória.

Em todos os testes a seguir, foram utilizados ladrilhos de resolução  $17 \times 17$ . Como o terreno tem resolução  $16.385 \times 16.385$ , tem-se uma árvore com dez níveis de profundidade e existem, portanto, aproximadamente 350.000 nós na árvore. O número de ladrilhos carregados em memória nestes testes foi limitado a 10.000, ou seja, aproximadamente 3% do número total de nós da árvore. Além disso, a predição era feita em um tempo de 125 milissegundos à frente do tempo atual.

### 5.3.1

#### Erro Médio e Máximo

Este teste tem o objetivo de analisar os erros projetados associados aos ladrilhos ao longo da navegação do caminho, fazendo uma comparação do comportamento deste erro com e sem o uso de predição. O erro é calculado conforme mostrado anteriormente na Seção 4.1.1. Um erro é coletado para análise sempre que o processo de divisão é interrompido, seja porque o ladrilho é uma folha (nesse caso o erro é zero), ou porque a tolerância foi atendida, ou ainda porque um ladrilho filho ou vizinho não está carregado. Este último caso é o de interesse deste teste, pois é o que provoca o não atendimento da tolerância especificada.

Inicialmente o erro médio é calculado fazendo a média dos erros observados em cada ladrilho. Com isso, pode-se traçar o gráfico da Figura 5.5, onde tem-se o erro médio ao longo do tempo de navegação do caminho.

Média dos Erros Médios	
<b>Sem Predição</b>	1,61
<b>Com Predição</b>	0,81

Tabela 5.5: Média ao longo do caminho do erro médio cometido em cada quadro

Na Tabela 5.5 pode-se observar a média do erro médio ao longo de todos os quadros visualizados. Observando esta tabela, já se pode perceber que o uso de predição apresentou um resultado melhor, pois conseguiu manter, em média, um erro abaixo da tolerância especificada de 1 pixel. Já o mesmo não aconteceu quando a predição não foi usada, caso em que o erro foi em média 2 vezes maior.

Além disso, pode-se observar no gráfico da Figura 5.5 que a falta de predição provocou um erro médio muito grande no trecho entre os pontos *c* e *d*, que é justamente um trecho de navegação rasante em alta velocidade. Neste trecho, a navegação é rápida, saindo de uma região muito plana para a uma região montanhosa. Na Figura 5.6 pode-se observar uma aproximação do gráfico da Figura 5.5, sendo os erros médios acima de 1 pixel descartados. Esta figura mostra que os erros médios com e sem predição foram bem próximos ao longo do caminho, exceto no trecho mencionado e no trecho entre os pontos *g* e *h*, onde ocorre também uma navegação de alta velocidade, mas com regiões menos montanhosas no final do trecho.

Porém, o erro médio não é a melhor medida de erro, pois não apresenta o que de pior aconteceu na visualização de um quadro. De maneira

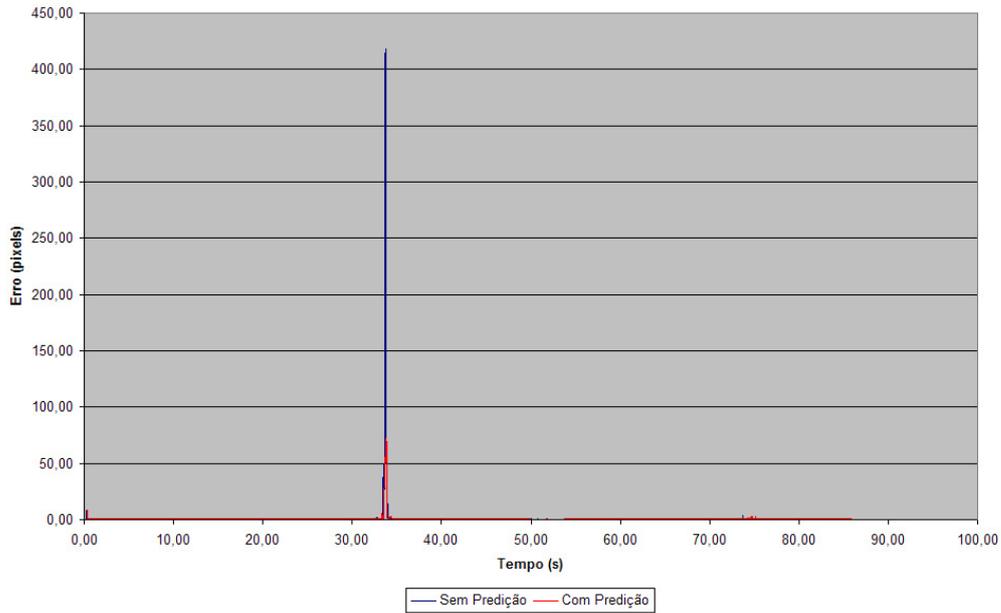


Figura 5.5: Erro médio em pixels, sem previsão e com previsão

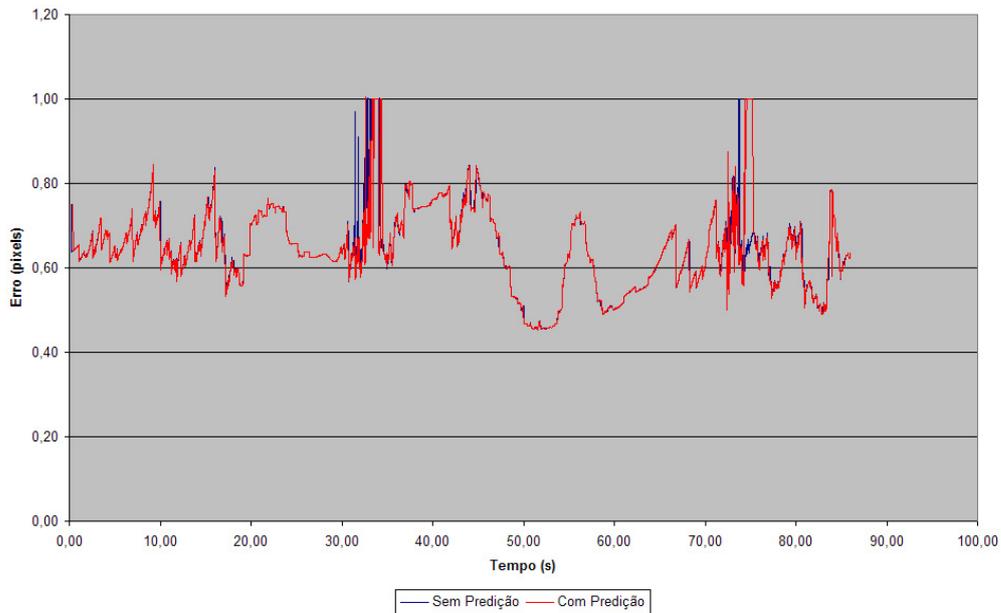


Figura 5.6: Aproximação da Figura 5.5 para mostrar maiores detalhes

semelhante, pode-se calcular o erro máximo observado no percurso da *quadtree*.

Média dos Erros Máximos	
<b>Sem Predição</b>	84,37
<b>Com Predição</b>	28,05

Tabela 5.6: Média ao longo do caminho do erro máximo cometido em cada quadro

Pela Tabela 5.6 pode-se observar que a média do erro máximo de todos os quadros foi da ordem de 3 vezes inferior com o uso da predição.

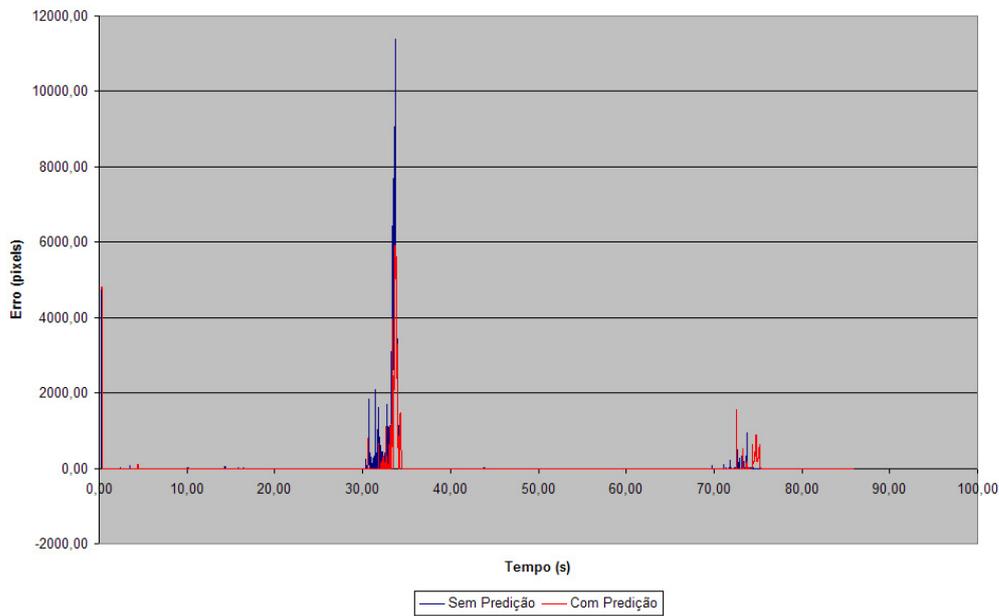


Figura 5.7: Erro máximo em pixels, sem predição e com predição

No gráfico ilustrado na Figura 5.7 pode-se observar que os trechos que apresentaram erros máximos elevados foram os mesmos, ou seja, os trechos entre os pontos *c* e *d* e entre *g* e *h*. No gráfico com aproximação da Figura 5.8 pode-se observar que agora os esses trechos ficam realmente piores sem o uso da predição. Além disso, os trechos entre os pontos *a* e *b* e entre *e* e *f* também são ruins quando não se usa predição. O primeiro apresenta uma navegação muito próxima ao chão, fazendo com que muitos ladrilhos novos sejam necessários entre quadros consecutivos. O segundo é um mergulho rápido, a partir de uma altura elevada, em direção a um vulcão com muitos detalhes.

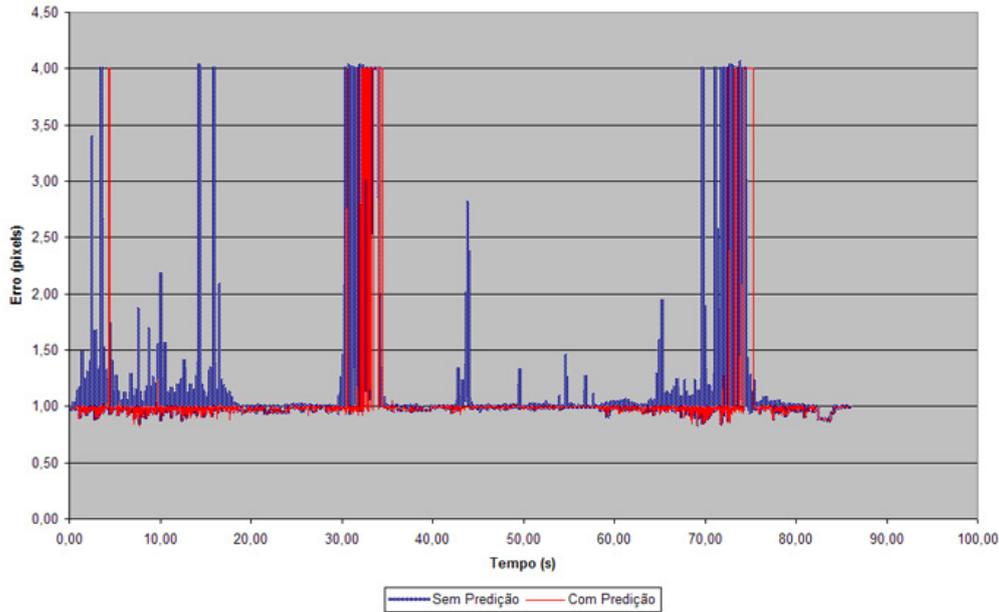


Figura 5.8: Aproximação da Figura 5.7 para mostrar maiores detalhes

### 5.3.2 Porcentagem de Ladrilhos Não Carregados

Este teste, como o anterior, tem o objetivo de avaliar se o uso de predição é vantajoso ou não. Para isso é feita, ao longo do caminho, a comparação entre o número de ladrilhos desenhados em um quadro e o número de ladrilhos que seria realmente desenhado se todos os ladrilhos necessários estivessem presentes em memória principal. Inicialmente, o caminho é percorrido ao longo do tempo, conforme nos testes anteriores, persistindo para cada quadro os parâmetros de câmera e o número de ladrilhos desenhados. Ao término do percurso, é feita a avaliação de quantos ladrilhos seriam realmente desenhados em cada quadro, utilizando os parâmetros de câmera armazenados. É calculada então uma porcentagem de quantos ladrilhos não foram desenhados dentre os ladrilhos que deveriam ser realmente desenhados.

Porcentagem Média de Ladrilhos Não Carregados	
<b>Sem Predição</b>	2,18 %
<b>Com Predição</b>	0,1 %

Tabela 5.7: Média ao longo do caminho da porcentagem de ladrilhos não carregados

A Tabela 5.7 representa a porcentagem média de ladrilhos não carregados ao longo do caminho de navegação. Fica clara a vantagem do uso de predição neste caso.

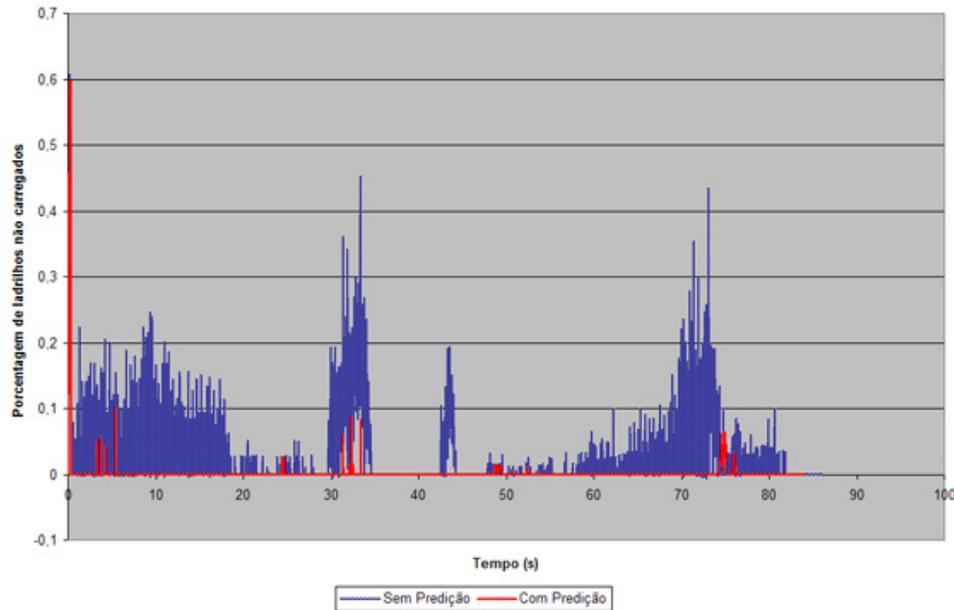


Figura 5.9: Porcentagem de ladrilhos não carregados sem predição e com predição

No gráfico da Figura 5.9, consideram-se os valores entre 0 e 1 como o intervalo de 0 a 100%. Nesta figura, pode-se observar um comportamento semelhante ao observado no gráfico da Figura 5.8 da seção anterior. O uso da predição apresenta-se mais vantajoso que no teste anterior, principalmente nos trechos entre os pontos *a* e *b*, *c* e *d*, *e* e *f* e entre *g* e *h*, pelos mesmos motivos mencionados anteriormente.