



Denis Hünich¹, <u>Ronny Tschüter²</u>, <u>Bert Wesarg^{1,3}</u>

¹ Information Services and High Performance Computing (ZIH), CIDS, TU Dresden ² Institute of Software Methods for Product Virtualization, German Aerospace Center (DLR) ³ GWT-TUD GmbH

Exploring Multi-threaded Communication Behavior of a Large-Scale CFD Solver with Vampir C/DS

15th International Parallel Tools Workshop 2024, Dresden, 2024 -09-19

Outline

Motivation

Distributed event loading

Evaluation

Racy message order

Conclusion & Outlook







Motivation



Exploring Multi-threaded Communication Behavior of a Large-Scale CFD Solver with Vampir 15th International Parallel Tools Workshop 2024, Dresden, 2024 -09-19 CIDS/ZIH, DLR, GWT-TUD





Performance Analysis of CODA



CODA

- Flow solver written by
 - ONERA
 - DLR
 - Airbus
 - Many other European partners

Coding

- User scenarios written in Python, solver core written in C++
- Hybrid parallelization, computation & communication overlap
- Spliss as linear solver



Courtesy: Immo Huismann

Performance Analysis of CODA: Current Status



- MPI provides different level of thread support
- CODA uses MPI + OpenMP threads with MPI_THREAD_MULTIPLE
 - Limited support in recent versions of performance analysis tools
- Initial performance analyses restricted to MPI_THREAD_FUNNELED
- Valuable first insights into applications performance, however
 - Varying the MPI level of thread support changes the application behavior

OMP thread 3:1 LU		y = A x (1134)				!\$10 yx	Jacobi ite
	y = A x (1134)					!\$10 yx	MPI_Wait
Aaster thread:2	$Cry = A \times (1134)$	$y = A \times (1134)$				MPIall y =	A x Ja
OMP thread 1:2	y = A x (1134)	y = A x (1134)				!\$omp bhpp:410 y	.x Ja
OMP thread 2:2	y = A x (1134)	y = A x (1134)				!\$omp bhpp:410 y =	A x Ja
DMP thread 3:2	$y = A \times (1134)$!\$omp bhpp:410 y =	Ax M
ster thread:3	$y = A \times (1134)$	$V = A \times (1134)$				My MPI Waitall yx	lacob
OMP thread 1:3	$y = A \times (1134)$	$y = A \times (1134)$				Isomp bart.hpp:410 yx	acob
MP thread 2:3 LU	$y = A \times (1134)$	$y = A \times (1134)$				Isomp bart.hpp:410 yx	lacob
P thread 3:3 LU	$y = A \times (1134)$	$y = A \times (1134)$!\$omp bart.hpp:410 yx	MPL
ter thread:4 LU	$Cry = A \times (1134)$	Md y = A x (1134)				My	V = AX
IP thread 1:4	$y = A \times (1134)$	$y = A \times (1134)$			-		$y = A \times [$
thread 2:4	$y = A \times (1134)$	$y = A \times (1134)$!\$0410	
thread 3:4 LU	$y = A \times (1134)$	$y = A \times (1134)$!\$0410	
r thread:5 LÚ	$y = A \times (1134)$	MPcv y = A x (1134)					y = A x
thread 1:5	$y = A \times (1134)$	$y = A \times (1134)$!\$om:410	y = A x
thread 2:5 LU	Cry = A x (1134)	$y = A \times (1134)$ y = A × (1134)				!\$om:410	y = A x
thread 3:5 LU	$y = A \times (1134)$	$y = A \times (1134)$ y = A x (1134)				!\$om:410	,
	$y = A \times (1134)$ $y = A \times (1134)$	y = A x (1134) y = A x (1134)				15011410	y = A x
er thread:6	$y = A \times (1134)$ $y = A \times (1134)$					Mann 110	
thread 1:6		$y = A \times (1134)$!\$om:410 }	
thread 2:6	$y = A \times (1134)$	$y = A \times (1134)$!\$om:410	
P thread 3:6 LU	y = A x (1134)	$y = A \times (1134)$ (33.54)				!\$om:410 y	
r thread:7 LU	y = A x (1134)	Mv y = A x (1134)				My MPII y	i
P thread 1:7	$y = A \times (1134)$	$y = A \times (1134)$			-	!\$omppp:410 y	
P thread 2:7 LU	y = A x (1134)	y = A x (1134)					- A X
P thread 3:7 LU	y = A x (1134)	y = A x (1134)				!\$omppp:410 y	
er thread:8	y = A x (1134)	y = A x (1134)				My yx	lacob
62.92450 s 62.9	2475 s 62.92500 s 62.925	25 s 62.92550 s 62.92575 s	62.92600 s 62.92	625 s 62.92650 s	62.92675 s	62.92700 s 62.92725 s	cob
x (1135)				\$omppp:410 y = A	LU	y = A x (1135)	IPI_V
A x (1135)				\$omppp:410 y = A	LU	y = A x (1135)	icobi
U	MPnd y = A	x (1135)				My MPI_Waitall y =	A x Ja
J	y = A x (1135)					<pre>!\$omp barrient.hpp:410 y =</pre>	Ax
U	··· • • (1125)	= A x (1135)					
	y = A x (1135) y					<pre>!\$omp barrient.hpp:410 y =</pre>	A X
U	Cry = A x (1135) y						A X A X
U y	Cry = A x (1135)					!\$omp barrient.hpp:410 y -	
U Y	Cry = A x (1135) y = A x (1135) A x (1135) y = A x (1135) y = A x (1135)				!\$omp	Isomp barrient.hpp:410 y = MPI_Waitall yx LU barriement.hpp:410 yx LU	
J S S S S S S S S S S S S S S S S S S S	Cry = A x (1135) y = A x (1135) A x (1135) y = A x (1135) y = A x (1135)				!\$omp !\$omp	I\$omp barrient.hpp:410 y = MPI_Waitall yx LU barriement.hpp:410yx LU	
y y y y	Cry = Ax (1135) Image: A grad and and and and a grad and and and and and and and and and a				!\$omp !\$omp	I\$omp barrient.hpp:410 y - MPI Waitall yx LU barriement.hpp:410yx LU barriement.hpp:410yx LU barriement.hpp:410yx LU	= A x
y y y y y	$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	Ax y = Ax (1135)			!\$omp !\$omp	Isomp barrient.hpp:410 y - MPI Waitall yx LU barriement.hpp:410yx LU barriement.hpp:410yx LU barriement.hpp:410yx LU barriement.hpp:410yx LU	- A x - A x
U	$ \begin{array}{c} C_{c,cr} y = A \times (1135) \\ \hline \\ A \times (1135) \\ = A \times (1135) \\ A \times (1135) \\ = A \times (1135) \\ = A \times (1135) \\ \hline \\ A \times (1135) \\ = A \times (1135) \\ \hline \\ C_{c,cr} y = A \times (1135) \\ \hline \\ \end{array} $	X y = A x (1135)			!\$omp !\$omp	İşomp barrient.hpp:410 y MPI Watall yx U barreent.hpp:410,x U barrement.hpp:410,x U barrement.hpp:410,x U jarrement.hpp:410,x U jarrement.hpp:410,x U	= A × = A × = A ×
) J	$\begin{array}{c} C_{v,ur} y = A \times (1135) \\ \hline \\ A \times (1135) \\ = A \times (1135) \\ Y = A \times (1135) \\ \hline \\ C_{v,ur} y = A \times (1135) \\ \hline \\ C_{v,ur} y = A \times (1135) \\ \hline \\ \hline \\ C_{v,ur} y = A \times (1135) \\ \hline \end{array}$	Δ x y = A x (1155) y = A x (1155)			!\$omp !\$omp	[\$comp barn.ent.hpp:410 y MPI watall yx [L] barriement.hpp:4109 x LJ barriement.hpp:4109 x LJ barriement.hpp:4109 x LJ barriement.hpp:4109 x LJ jbarriement.hpp:4109 x LJ y [so410 y [so410	A x A x A x A x A x A x A x
U U	$\begin{array}{c} C_{\rm m}ry = A \times (1135) \\ A \times (1135) \\ = A \times (1135) \\ \\ C_{\rm m}ry = A \times (1135) \\ \\ \\ C_{\rm m}ry = A \times (1135) \\ \\ \\ \end{array}$	x y - A x (1135) y - A x (1135) y - A x (1135)			!\$omp !\$omp	Išomp barrient.hpp:410 y - MPI, Watall y - xx UU barriement.hpp:410 yx UU barriement.hpp:410 x UU barriement.hpp:410 x UU barriement.hpp:410 x UU barriement.hpp:410 x UU ibarriement.hpp:410 x UU ibarrie	× A = × A = × A =
J.	$\label{eq:constraints} \begin{split} & (C_{xy} - A \times (1155) \\ & = C_{xy} - A \times (1155) \\ & (C_{xy} - Y + A \times (1135)) \\ & (C_{xy} - Y + (C_{xy} - Y$	$y = A \times (1135)$!\$omp !\$omp	Homp barnent.htp:410 y Mor Waitall y k Barnement.hpp:410 u U barnement.hpp:410 u U barnement.hpp:410 u U barnement.hpp:410 u U barnement.hp:410 u U barnement.hp:410 u U iso410 y iso410 y iso410 y iso410 y iso410 y iso410 y iso410 y iso410 y	$= A \times$
U U	$\begin{array}{c} (C,ry \rightarrow x(1155) \\ \hline \\ A(1155) \\ a(1155) \\ Y \rightarrow X(1135) \\ A(1155) \\ Y \rightarrow X(1135) \\ A(1155) \\ Y \rightarrow X(1135) \\ C,r,y \rightarrow X(1135) \\ \hline \\ C,r,y \rightarrow X(1135) \\ C,r,y \rightarrow X(1135) \\ \hline \\ C,r,y \rightarrow X(1135) \\ \hline \\ Helyneev \\ Y \rightarrow X(1135) \end{array}$	$y = A \times (1135)$!\$omp !\$omp	Išomp barrient.hpp:410 y - MP; Vatali y - ML barriement.hpp:410 y LU iSom10 y ISo410 y ISom10 y ISo410 y ISom	$= A \times$
U U	$\begin{array}{c} C_{\rm m}ry = A \times (1115) \\ A \times (1125) \\ A \times (1125) \\ A \times (1135) \\ Y = A \times (1135) \\ A \times (1135) \\ Y = A \times (1135) \\ C_{\rm m}ry = A \times (1135) \\ MH \ Hecv \\ Y = A \times (1135) \\ MH \ Hecv \\ Y = A \times (1135) \\ Y = A \times (11$	xx y = Ax (1135) $y = Ax (1135)$ $y = Ax (1135)$ $y = Ax (1135)$ $y = Ax (1135)$!\$omp !\$omp	Išomp barrient.hpp:410 y - MPI Vatall y - KU barriement.hpp:410 y ISo410 y ISo410 y ISo410 y ISompp:410 y ISompp:410 y	$= A \times$
	$\begin{array}{c} (C_x \gamma = A \times (1155) \\ = A \times (1135) \\ = A \times ($	$\begin{array}{c c} x & y = A \times (1135) \\ \hline y = A \times (1135) \\ \hline y = A \times (1135) \\ \hline \end{array}$!\$omp !\$omp	Homp barriett.htp:410 y MPI Watall y:x LU MPI watall y:x LU barriement.htp:410 x LU barriement.htp:410 x LU barriement.htp:410 x LU barriement.htp:410 x LU iso410 y Iso410 y Iso410 y Iso410 y	$= A \times$
	$\begin{array}{c} C_{\rm m}ry = Ax(1155) \\ Ax(1125) & y = Ax(1135) \\ Ax(1135) & y = Ax(1135) \\ Ax(1135) & y = Ax(1135) \\ Ax(1135) & y = Ax(1135) \\ C_{\rm m}ry = Ax(1135) \\ PPPrevol \\ y = Ax(1135) \\ y = Ax(115) \\ y = Ax$	$y = A \times (1135)$!\$omp !\$omp	Išomp barrient.hpp:410 y - MPI vatali y - WI barriement.hpp:410 yx UU iSo410 y iSo410 y iSo410 y iSo410 y iSom.pp.410 y iSompp.410 y iSomp	$= A \times$
.U .U	$ \begin{array}{c} C_{x,Y} = A \times (1155) \\ C_{x,Y} = A \times (1155) \\ \hline \\ C_{x,Y} = A \times (1155) \\ \hline \\ C_{x,Y} = A \times (1155) \\ \hline \\ P = A \times (1155) \\ P = A \times (1155) \\ \hline \\ P = A \times (1155) \\ P = A \times (1155) \\ \hline \\ C_{x,Y} = A \times (1155) \\ \hline \\ \end{array} $	$y = A \times (1135)$!\$omp !\$omp	Homp barrient.hip-410 y MP Watall y:x LU MP watall y:x LU barriement.hip-410 x LU (So410 y (So410 y (Sompp.410 y (Sompp.410 y (Sompp.410 y (Sompp.410 y (Sompp.410 y (Sompp.410 y	$A \times$
U U	$ \begin{array}{c} (C_{11}y \rightarrow A \times (115) \\ = A \times (115) \\ = A \times (115) \\ = A \times (1135) \\ = A \times ($	$\begin{array}{c} x y = A x \ (1135) \\ z = A x \ (1135) \ (113$!\$omp !\$omp	Išomp barriett.hpp:410 y-m U/ MPI, Vatall y-m LU barriement.hpp:410 yx LU barriement.hpp:410 y 150410 Y 150410 Y 150410 Y 15010 Y 150410 Y 150	$= A \times$
U U	$\begin{array}{c} C_{x,y} - x \times (1155) \\ = X \times $	$y = A \times (1135)$!\$omp !\$omp	Homp barrient.hip-410 y Mar Waitall y K Mar Waitall y K Mar Waitall y K Mar Waitall y K Barriement.hip/410 y K barriement.hip/410 y K iso410 y So410 y Iso410 y Iso410 y So410 iso410 y Iso410 Isom	$= A \times$
	$\begin{array}{c} C_{x,Y} - x \times (115) \\ = X \times (113) \\ = X \times (113) \\ = X \times (113) \\ \hline \\ C_{x,Y} = X \times (113) \\ \hline \\ C_{x,Y} = X \times (113) \\ \hline \\ C_{x,Y} = X \times (113) \\ \hline \\ Y = X \times (113)	$\begin{array}{c c} x & y = A \times (1135) \\ \hline y = A \times (1135) \\ y = A \times $!\$omp !\$omp	Homp barnent.htp:410 y.m. LU MPI_Watabil Y.m. LU LU MPI_watabil Y.m. LU LU barnement.htp:410 y.m. LU LU barnement.htp:410 y.m. LU LU barnement.htp:410 y.m. LU LU ISo410 Y LSo410 Y ISo410 Y LSo410 Y ISompp.410 y= LSompp.410 Y ISompp.410 Y LSomp410 Y ISomp	$A \times A \times$
J.	$\begin{array}{c} C_{x,Y} - x \times (115) \\ = X \times (113) \\ = X \times (113) \\ = X \times (113) \\ \hline \\ C_{x,Y} = X \times (113) \\ \hline \\ C_{x,Y} = X \times (113) \\ \hline \\ C_{x,Y} = X \times (113) \\ \hline \\ Y = X \times (113)	$\begin{array}{c} x y = A x \ (1135) \\ z = A x \ (1135) \ (113$!\$omp !\$omp	Homp barrient.hip-410 y Mar Waitall y K Mar Waitall y K Mar Waitall y K Mar Waitall y K Barriement.hip/410 y K barriement.hip/410 y K iso410 y So410 y Iso410 y Iso410 y So410 iso410 y Iso410 Isom	$A \times A \times$
	$\begin{array}{c} C_{\rm er} y = A \times (1115) \\ A \times (1125) & y = A \times (1125) \\ A \times (1135) & y = A \times (1135) \\ A \times (1135) & y = A \times (1135) \\ A \times (1135) & y = A \times (1135) \\ C_{\rm er} y = A \times (1135) \\ C_{\rm er} y = A \times (1135) \\ C_{\rm er} y = A \times (1135) \\ Y = A \times (1135$	$\begin{array}{c c} x & y = A \times (1135) \\ \hline y = A \times (1135) \\ y = A \times $!\$omp !\$omp	Homp barnent.htp:410 y.m. LU MPI_Watabil Y.m. LU LU MPI_watabil Y.m. LU LU barnement.htp:410 y.m. LU LU barnement.htp:410 y.m. LU LU barnement.htp:410 y.m. LU LU ISo410 Y LSo410 Y ISo410 Y LSo410 Y ISompp.410 y= LSompp.410 Y ISompp.410 Y LSomp410 Y ISomp	= A x = A x
	$ \begin{array}{c} C_{x,Y} - x \times (1155) \\ = 7 \times$	$y = A \times (1135)$!\$omp !\$omp	Išomp barrient.hpp.410 y MP, Vatali y w U MP, Vatali y w U barriement.hpp.410 y x U barriement.hpp.410 y x U barriement.hpp.410 y x U barriement.hpp.410 y x U iso410 y iso410 y iso410 y iso410 y isomp	- A x - A x
	$ \begin{array}{c} C_{x,Y} - x \times (1155) \\ = 7 \times$	$\begin{array}{c} x y = A \times (1135) \\ \hline y = A \times (1135) \\ z = A \times (1135) \\ = A \times (1135) \\ z = A \times (113$!\$omp !\$omp	Homp barrient.hip-410 y MP Variall y y MP Variall y x MP variall y x MP variall y x barriement.hip-410 y xx U barriement.hip-410 y xx U barriement.hip-410 y xx U iso410 y iso410 y iso410 y iso410 y iso410 y iso410 y isom.pp.410 y y iso410 y isom410 y isom410 y isom	- A x - A x
	$ \begin{array}{c} C_{x,y} - x \times (1155) \\ = X \times (1155) \\ \hline \\ C_{x,y} - x \times (1135) \\ \hline \\ C_{x,y} - x \times (1135) \\ \hline \\ C_{x,y} - x \times (1135) \\ \hline \\ P + x \times (1135) \\ P + x \times (1135) \\ \hline \\ P + x \times (1135) \\ P + x \times (1135) \\ \hline \\ P + x \times (1135) \\ P + x \times (1135) \\ \hline \\ P + x \times (1135) \\ P + x \times (1135) \\ \hline \\ P + x \times (1135) \\ P + x \times (1135) \\ \hline \\ P + x \times (1135) \\ P + x \times (1135) \\ \hline \\ P + x \times (1135) \\ P + x \times (1135) \\ \hline \\ P + x \times (1135) \\ P + x \times (1135) \\ \hline \\ P +$	$\begin{array}{c} x y = A \times (1135) \\ \hline y = A \times (1135) \\ z = A \times (1135) \\ = A \times (1135) \\ z = A \times (113$			ISomp ISomp ISomp	Hšomp barriett.hbp:410 y MP, Vatall y MI MP, Vatall y MI barrement.hpp:410 yx U barrement.hpp:410 yx U barrement.hpp:410 yx U barrement.hpp:410 y y i5o10 Y Y i5omp.p.p.10 y Y i5ompp.410 y Y i5ompp.410 y Y i5omp	- A x - A x
1	$\begin{array}{c} C_{x,Y} - x \times (115) \\ \hline \\ = X \times (115) \\ = X \times (113) \\ \hline \\ = X \times (113) \\ \hline \\ \\ C_{x,Y} - x \times (113) \\ \hline \\ C_{x,Y} - x \times (113) \\ \hline \\ C_{x,Y} - x \times (113) \\ \hline \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ $	$\begin{array}{c} x y = A \times (1135) \\ \hline y = A \times (1135) \\ z = A \times (1135) \\ = A \times (1135) \\ z = A \times (113$			ISomp ISomp ISomp ISomp ISom ISom	Homp barrient.hipp:410 y Marriment.hipp:410 w. U barriement.hipp:410 w. U iso410 y iso410 iso	= A x = A x

R. Tschüter, I. Huismann, B. Wesarg and M. Knespel, "Performance Analysis of the CFD Solver CODA -Harnessing Synergies between Application and Performance Tools Developers," 2022 IEEE/ACM Workshop on Programming and Performance Visualization Tools (ProTools)

5

Performance Analysis of CODA: Current Status



- Limitations w.r.t. handling the MPI_THREAD_MULTIPLE level of thread support
 - Message matching does not work
 - Excessive increase in time to load trace data
 - Severely restricted insight into communication patterns

6

Distributed event loading



Exploring Multi-threaded Communication Behavior of a Large-Scale CFD Solver with Vampir 15th International Parallel Tools Workshop 2024, Dresden, 2024 -09-19 CIDS/ZIH, DLR, GWT-TUD

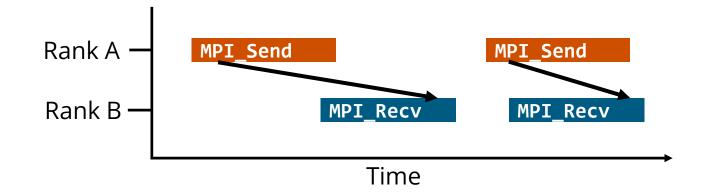


MPI point-to-point message matching

Need pairwise event matching to derive upper bound of message duration

MPI messages consists of a *send* and *receive* event in the same *envelop*

Timestamp of sends and receives determine message matching order

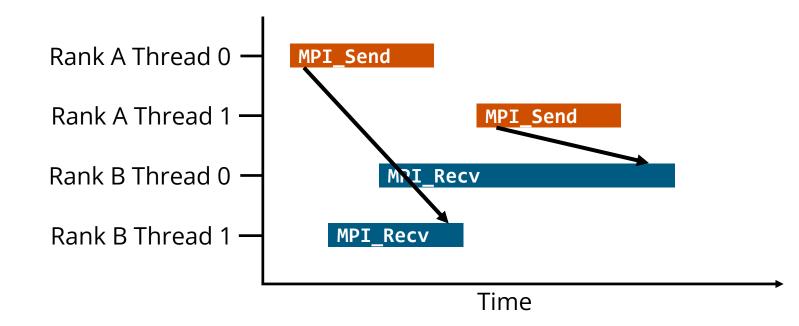






MPI point-to-point message matching MPI_THREAD_MULTIPLE

Must obey message order across all threads of a rank

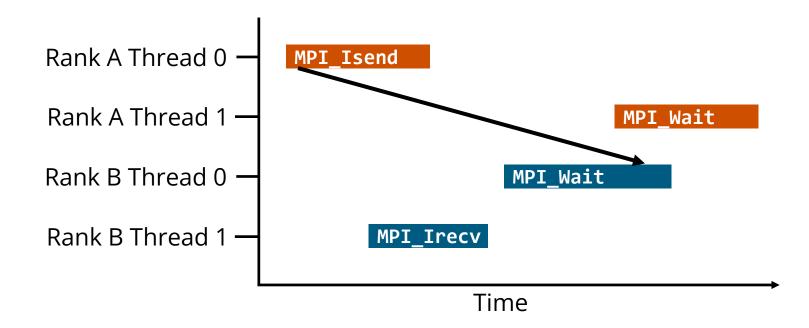








MPI point-to-point message matching MPI_THREAD_MULTIPLE & non-blocking

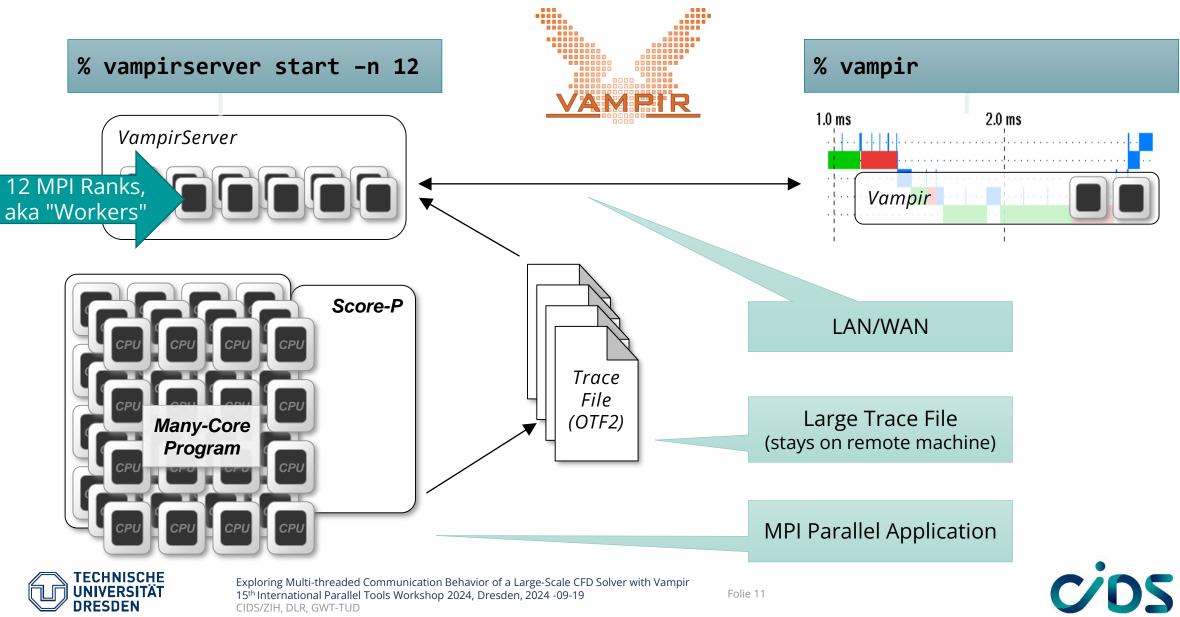








Vampir



CIDS/ZIH, DLR, GWT-TUD

Vampir Distributed event processing

Storing events for one location (thread, accelerator stream, ...) in one worker ≻Open each file at most once

Randomly distribute locations to workers

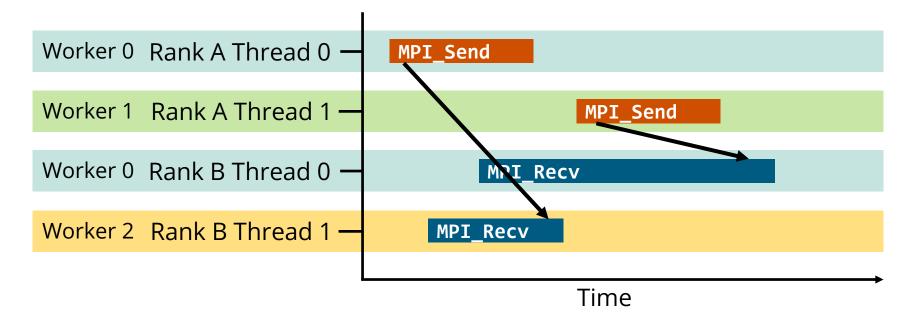
➢Balances memory and compute load on average







Vampir Distributed event processing and MPI_THREAD_MULTIPLE



Distribution destroys event order across threads of one rank

>Need to process all ordered events of a process by one worker





Vampir

Constrains:

- Do not interfere with the visualization (balance memory and compute)
- Avoid opening event files by multiple workers

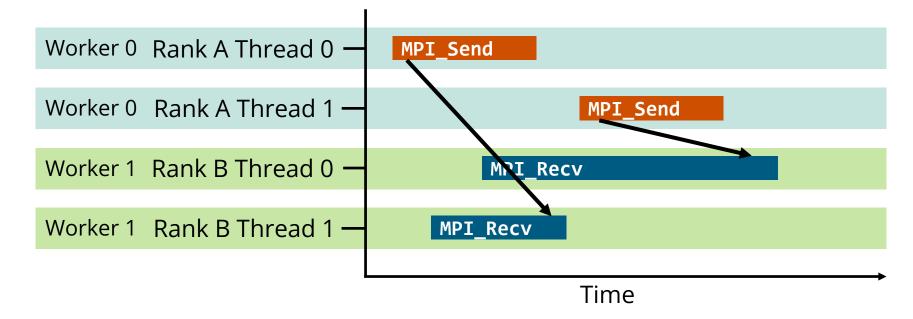
Ideas:

- 1. Keep distribution and send MPI events around
- Implementing a parallel discrete event simulation
- 2. Keep distribution, but let workers read all necessary event files
- Reading the same event file by multiple workers
- 3. Change distribution at load time only
- Restores distribution via message across workers





Vampir Load time distribution



Folie 15

➢Reduces load balance at load time







Evaluation



Exploring Multi-threaded Communication Behavior of a Large-Scale CFD Solver with Vampir 15th International Parallel Tools Workshop 2024, Dresden, 2024 -09-19 CIDS/ZIH, DLR, GWT-TUD





Measurement environment:

- CIDS/ZIH HPC-Cluster Barnard with 2 x Intel Xeon Platinum 8470 (52 cores) @ 2.00 GHz per node
- Compiler: GCC 13.2

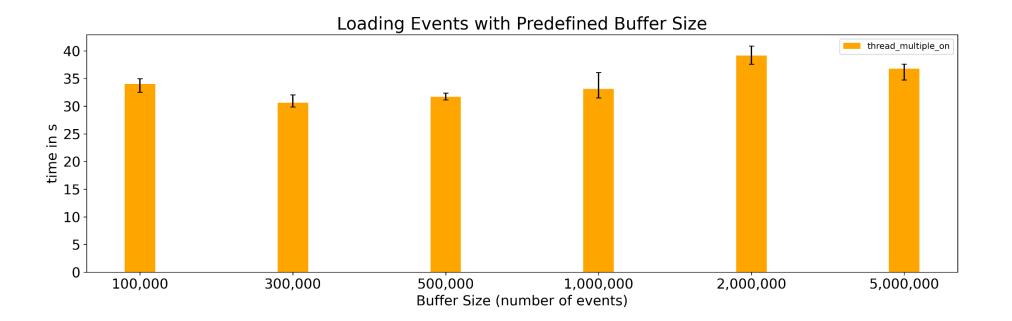
Measurement parameter:

- Vampir versions:
- Current version of Vampir (master)
- Vampir with MPI_THREAD_MULTIPLE support turned on (*thread_multiple_on*)
- Vampir with MPI_THREAD_MULTIPLE support turned off (*thread_multiple_off*)
- Sample size is 5
- CODA Trace:
 - Application that used MPI_THREAD_MULTIPLE
 - 64 ranks with 4 threads each
 - 2.8 GiB





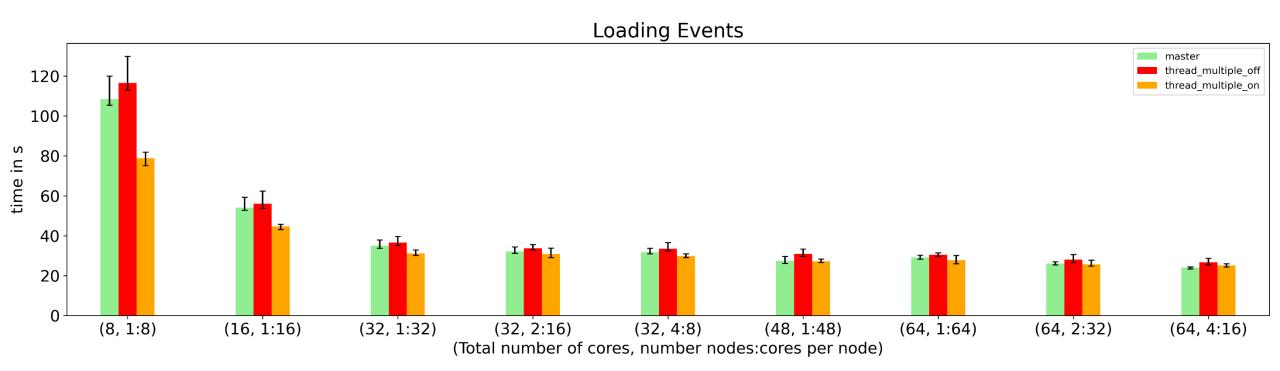
MPI_THREAD_MULTIPLE support turned on and 32 MPI ranks on one node Different buffer sizes to distribute events for balanced worker/location ratio







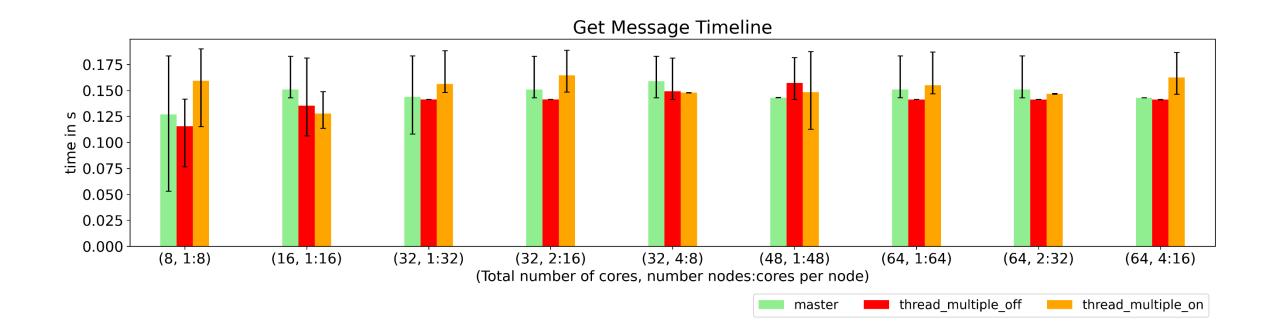
Buffer size of 300,000 used for thread_multiple_on





Folie 19



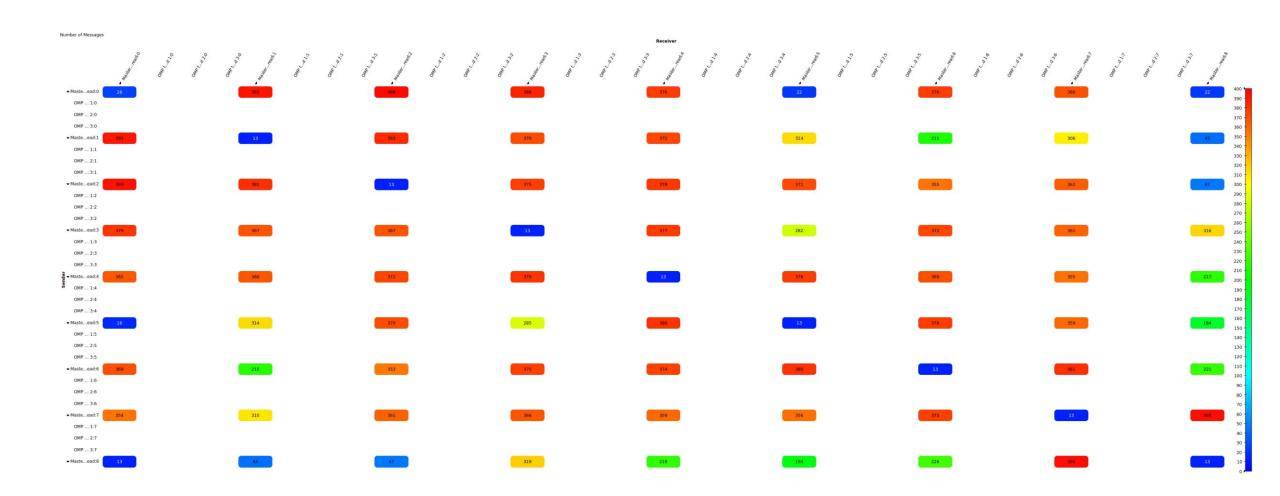








Vampir Communication Matrix (Master)



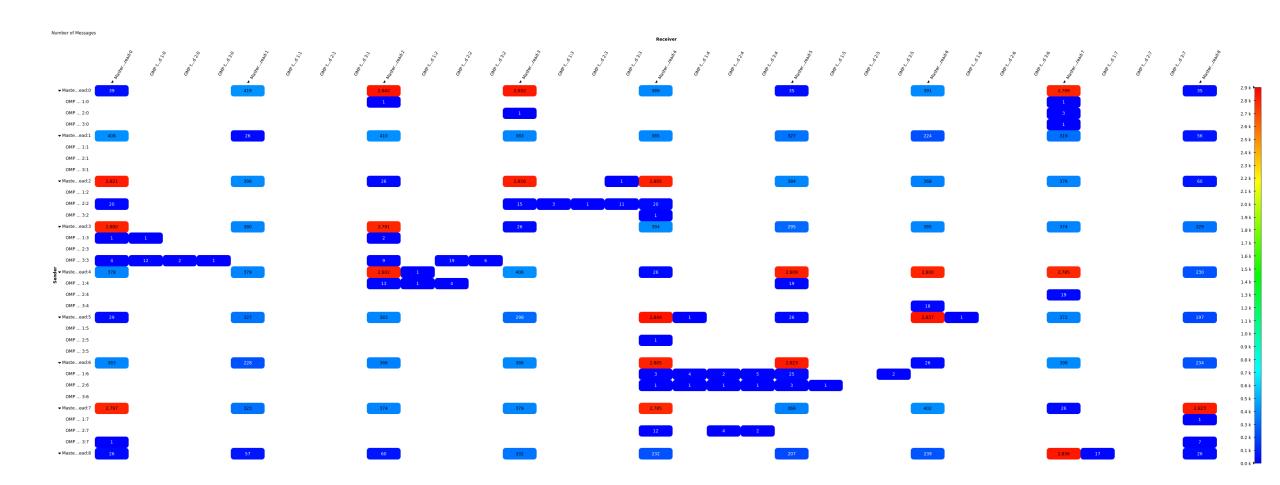


Exploring Multi-threaded Communication Behavior of a Large-Scale CFD Solver with Vampir 15th International Parallel Tools Workshop 2024, Dresden, 2024 -09-19 CIDS/ZIH, DLR, GWT-TUD

Folie 21



Vampir Communication Matrix (*thread_multiple_on*)





Folie 22



Racy message order

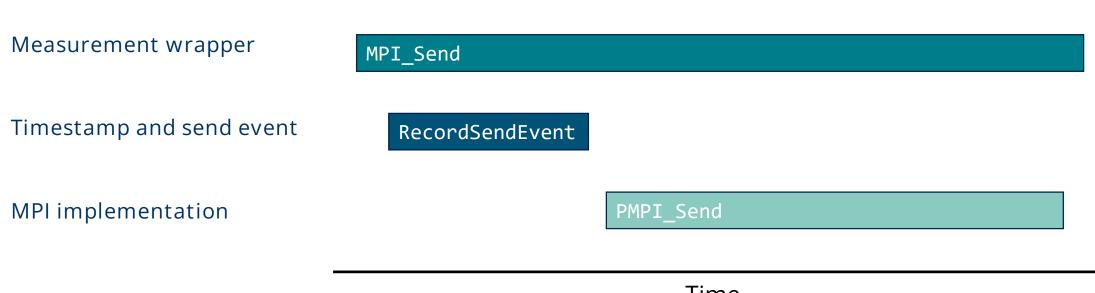


Exploring Multi-threaded Communication Behavior of a Large-Scale CFD Solver with Vampir 15th International Parallel Tools Workshop 2024, Dresden, 2024 -09-19 CIDS/ZIH, DLR, GWT-TUD



Determining message order

Timestamps of MPI calls determine message order











Level of thread support in

Level of thread support	Ruling	Serialized on MPI call level
MPI_THREAD_SINGLE	Only one thread will execute.	Yes
MPI_THREAD_FUNNELED	Only the thread that called MPI_Init_thread will make MPI calls.	Yes
MPI_THREAD_SERIALIZED	Only one thread will make MPI library calls at one time.	Yes
MPI_THREAD_MULTIPLE	Multiple threads may call MPI at once with no restrictions.	No

Folie 25

➤Serializations happens inside MPI implementation







Racy message order

Avoid:

- —Force serialization in the measurement
- —Change application to use different tags (MPI_TAG_UB)
- -Needs help of MPI implementation

Detect:

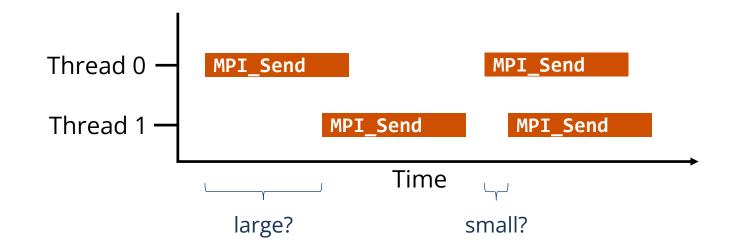
- —Are there any two messages with the same envelop on different threads at all?
- —Are there any two MPI post calls with the same envelop on different threads which overlap?
- —Is the difference between two MPI post calls with the same envelop on different threads "small" enough to permit a race?





Folie 26

Racy message order Inter Message Duration



Global minimum value presented to user

User needs to interpret value







Conclusion & Outlook



Exploring Multi-threaded Communication Behavior of a Large-Scale CFD Solver with Vampir 15th International Parallel Tools Workshop 2024, Dresden, 2024 -09-19 CIDS/ZIH, DLR, GWT-TUD



Conclusion & Outlook

Conclusion Vampir able to visualize applications with any level of thread support Acceptable overhead when loading No overhead after loading

Outlook

Overhead only needed if level of thread support exceeds MPI_THREAD_FUNNELED > Measurement annotates trace with level of thread support Improve racy message order by checking MPI call overlap Opens new analysis possibilities for other paradigms

Folie 29



