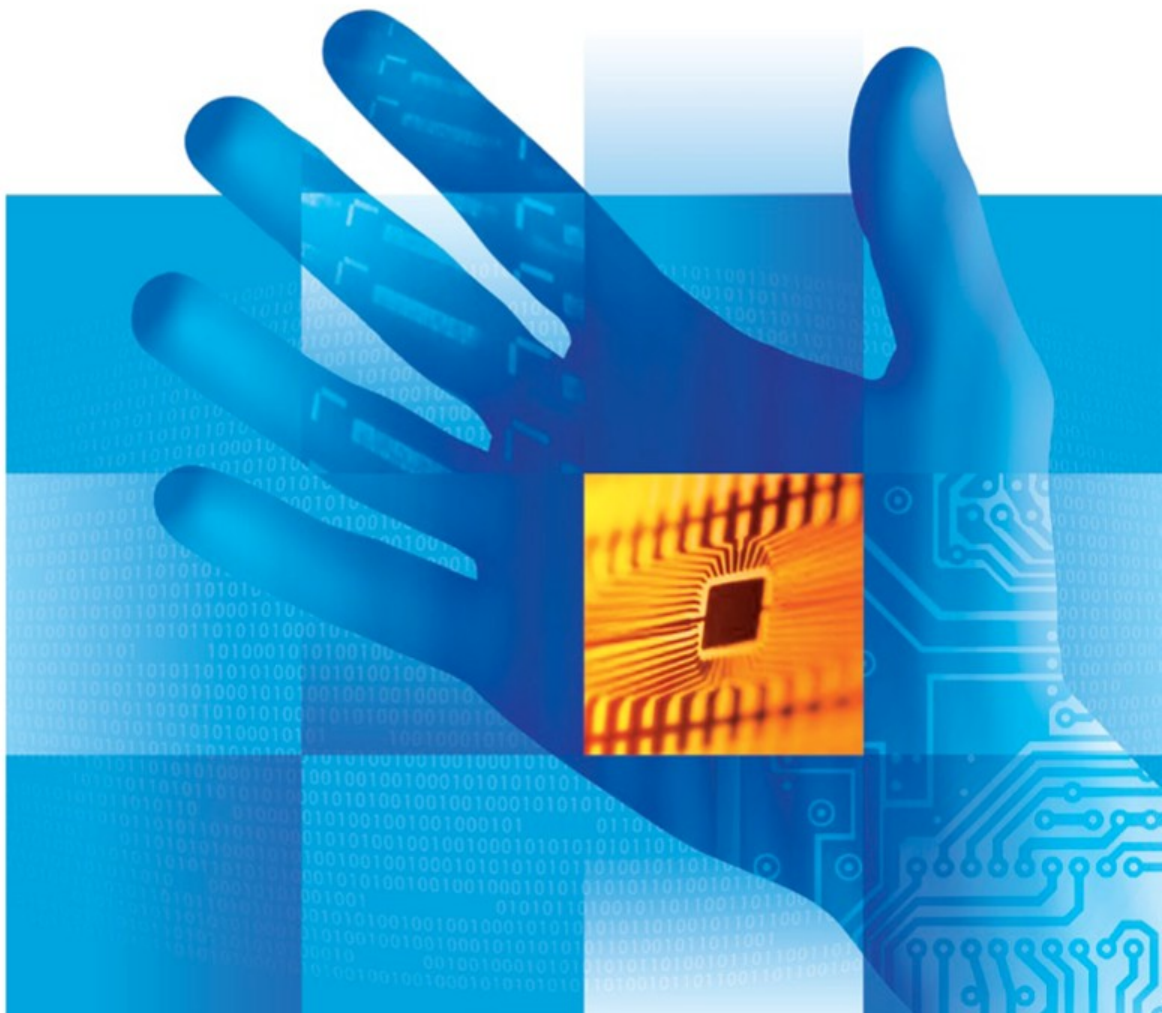




Matematika, supercomputing a řešení reálných úloh

Zdeněk Dostál
Národní
superpočítačové
Centrum IT4I
a
Katedra aplikované
Matematiky
FEI VŠB-TU Ostrava

ŠKOMAM 2014





Osnova

1. Supercomputing a Národní superpočítačové centrum
2. Charakteristiky superpočítačových algoritmů – škálovatelnost
3. Příklad: eliminace a superpočítačové algoritmy
4. Teoretické výsledky
5. Příklady





Superpočítače

Wiki: „A **supercomputer** is a computer at the frontline of contemporary processing capacity – particularly speed of calculation.“

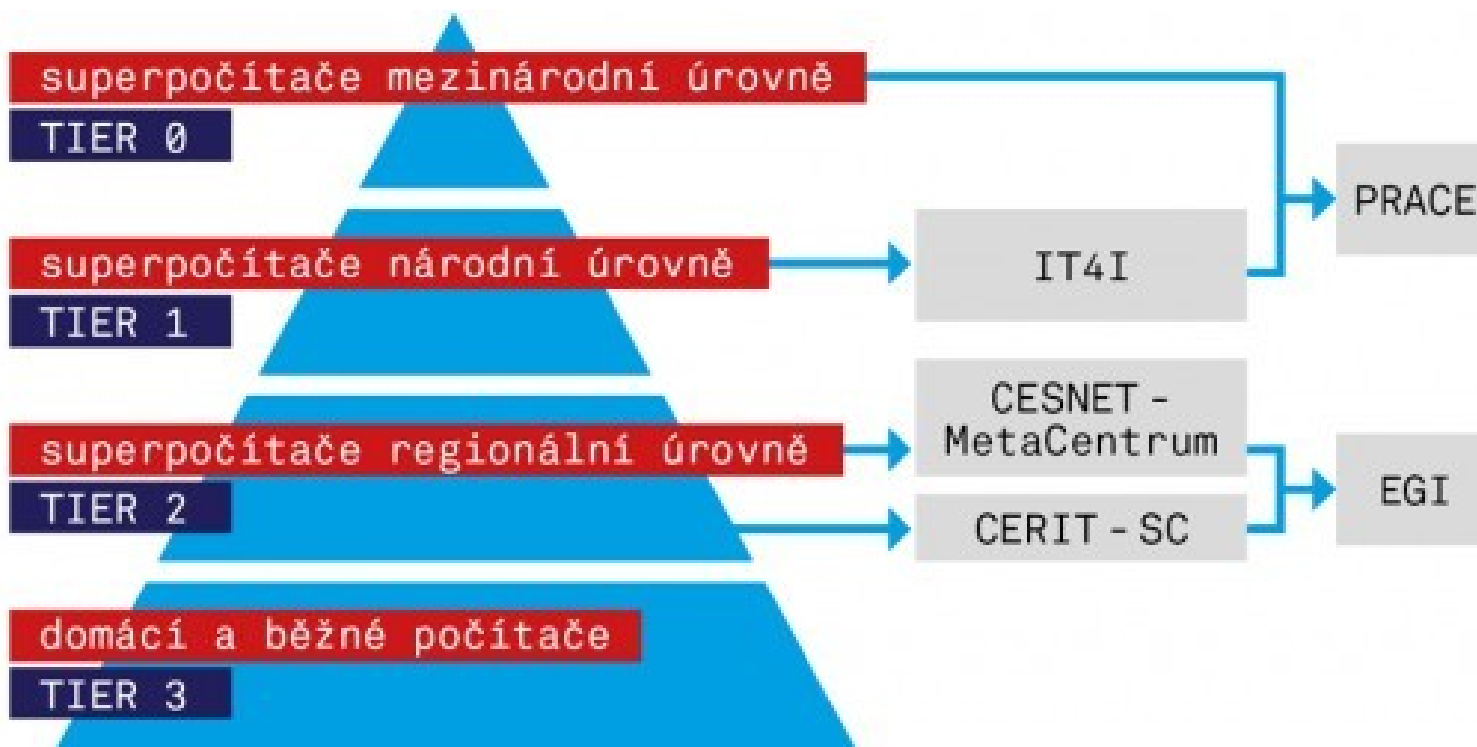
Year	Supercomputer	<u>Peak speed (Rmax)</u>	Location
2008	<u>IBM Roadrunner</u>	1.026 PFLOPS	<u>Los Alamos</u> ,
2009	<u>Cray Jaguar</u>	1.759 PFLOPS	<u>Oak Ridge</u> , USA
2010	<u>Tianhe-I</u>	2.566 PFLOPS	<u>Tianjin</u> , <u>China</u>
2011	<u>Fujitsu K computer</u>	10.51 PFLOPS	<u>Kobe</u> , <u>Japan</u>
2012	<u>Cray Titan</u>	17.59 PFLOPS	<u>Oak Ridge</u> , USA
2013	<u>NUDT Tianhe-2</u>	33.86 PFLOPS	<u>Guangzhou</u> , China
2020	???	~1000 PFLOPS	???

1 PFLOP=1e15 operací za vteřinu





Klasifikace superpočítačů





Národní superpočítačové centrum

- VŠB – OU, SU, UGN, ČVUT
- Malý cluster (3312 jader, 0.1PFLOPS, 0.1 PB paměť)
- 8 výzkumných programů
- Velký cluster ~10x rychlejší - 2015





Př. 1: Modelování povodní

Cílem je detailně předpovědět průběh povodní
(sběr dat, modelování povrchového proudění,
vizualizace, ...)

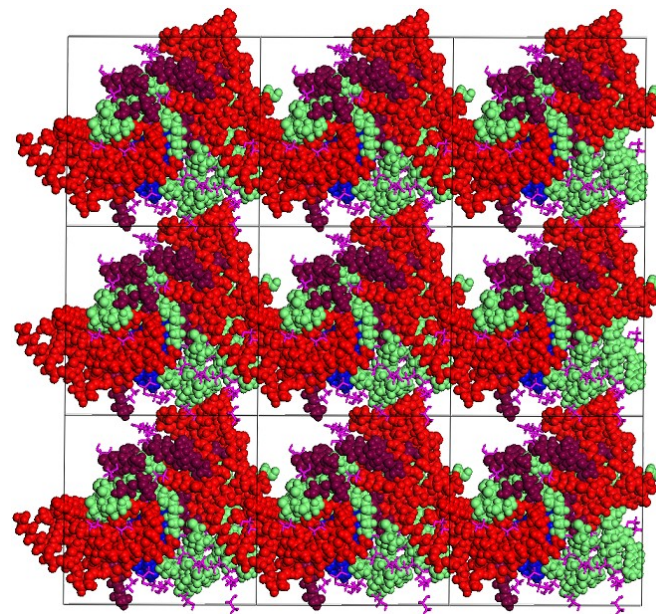
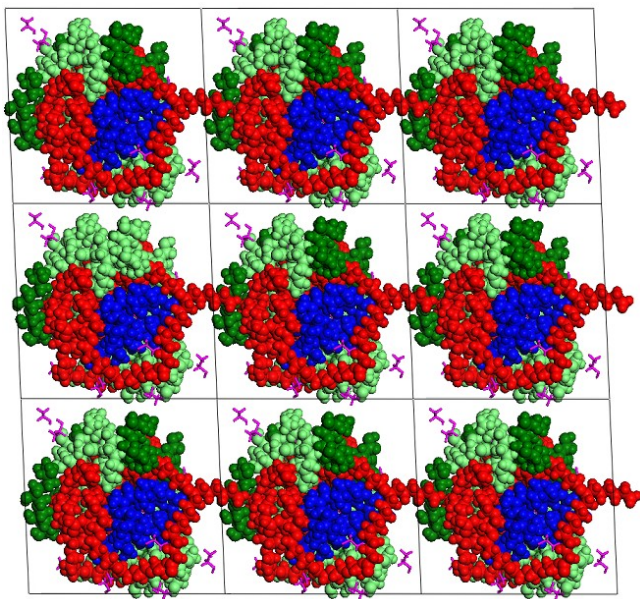


(obr. V. Vondrák VP1)



Př. 2: Počítačové modelování nosičů léků a vývoj léků

Cílem je předvídat vlastnosti nosičů léků

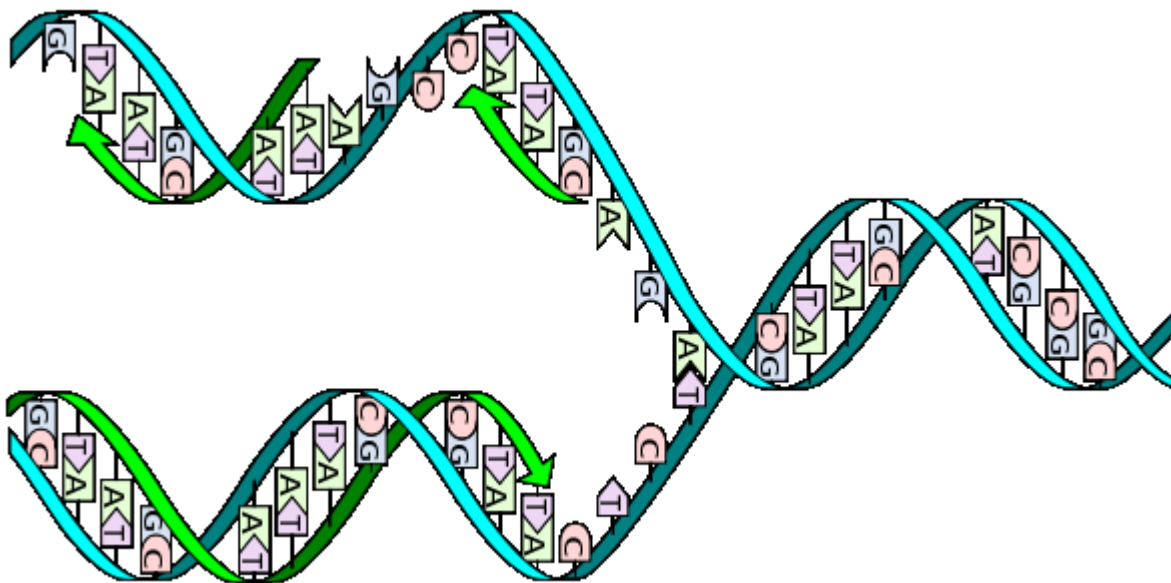


(obr. J. Pištora, VP4)



Př. 3: Porovnávání DNA

Cílem je zjistit genetické odchylky, náchylnost k nemocem, reakci na léky, otcovství ...

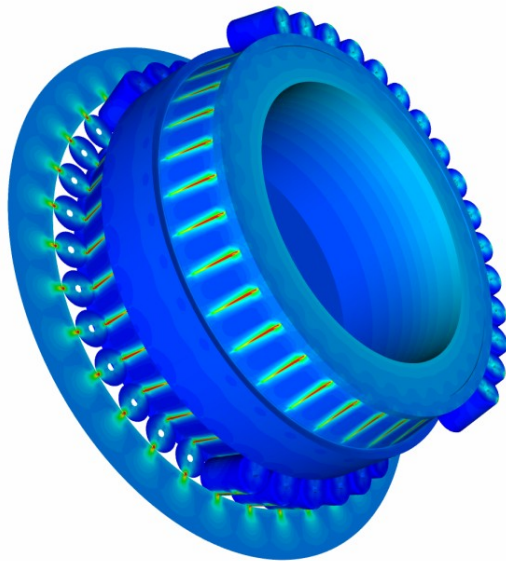


Problémy: Co je vzdálenost dvou řetězců? Jak ji spočítat, když máme jen části řetězců? ...



Př. 4: Inženýrské problémy

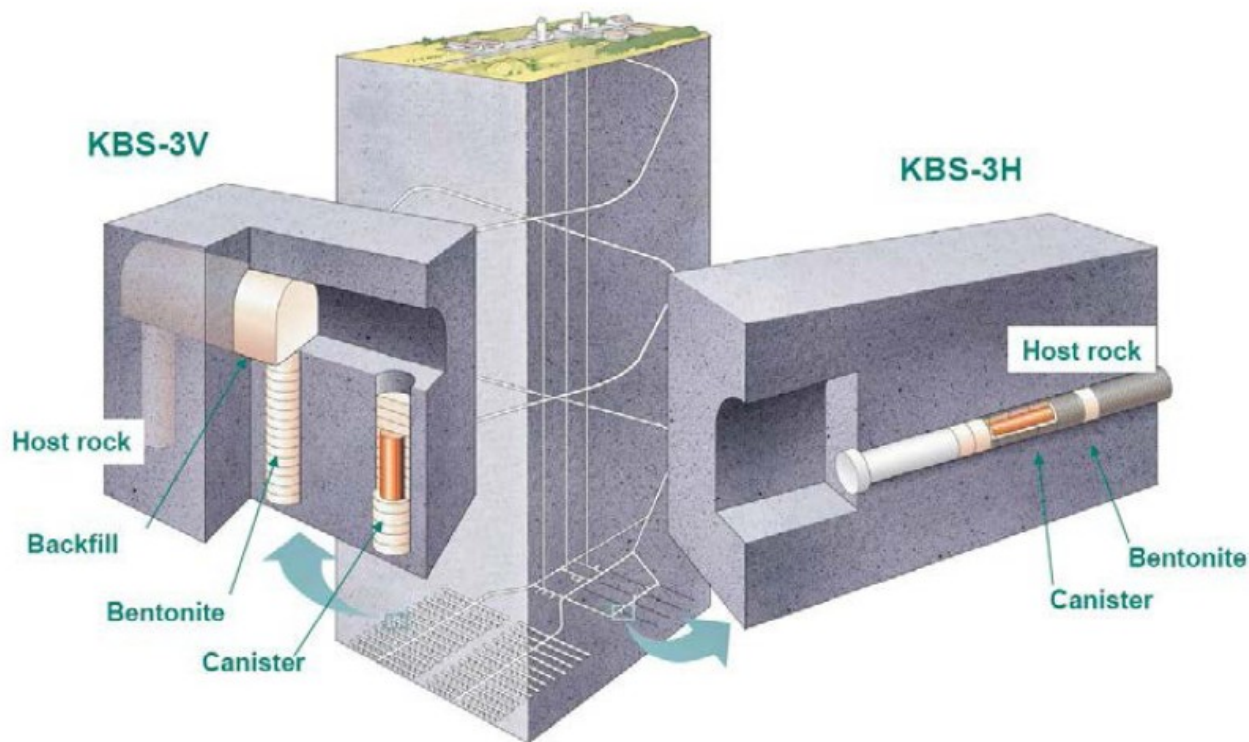
- Cílem je popsat namáhání a deformaci detailů strojních konstrukcí





Př. 5: Ukládání radioaktivních odpadů

Cílem je předpovědět vliv radioaktivního odpadu na okolní prostředí na 100 a více let dopředu



(obr. R. Blaheta, VP2)



Cíle matematického modelování

Najít cestu **od vstupních informací** (tvar tělesa, působící síly, zdroje teploty, uchycení, materiálové vlastnosti) **k požadovaným výstupním informacím** (deformace tělesa, posunutí, napětí, rozložení teplot)





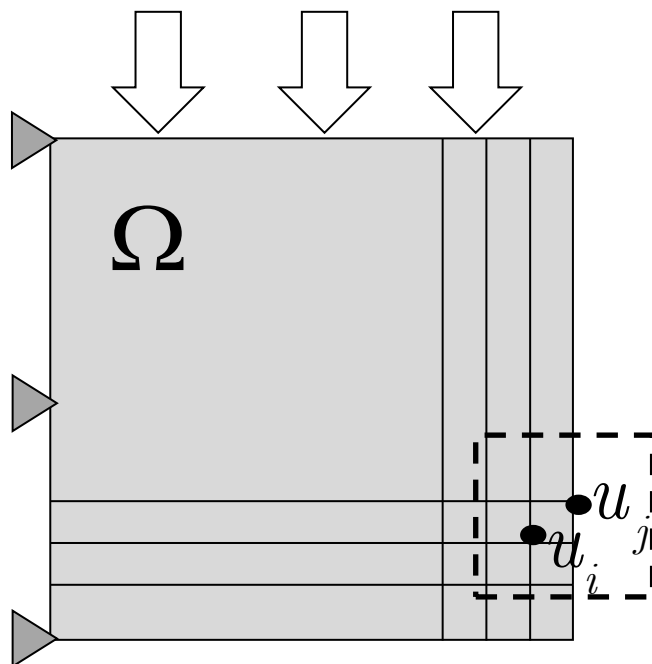
Postup matematického modelování

1. Pozorování a formulace obecných zákonitostí (rovnice rovnováhy, variační principy, oblast-hranice)
2. Zjednodušení pomocí vhodných předpokladů (lineární pružnost, zanedbání tření, ...), spojitá formulace
3. Redukce na problém konečné dimenze
4. Redukce na posloupnost řešení soustav standartních problémů (lineárních rovnic)
5. Řešení posloupnosti problémů (rovníc)
6. Zpracování výsledků (počítačová grafika)





Příklad: Deformace tělesa





Problémy “čisté” matematiky

1. Existence řešení
2. Jednoznačnost, charakteristika fyzikálního řešení
3. Analýza citlivosti (tvar oblasti, konstanty), co je malá změna?





Adaptace algoritmů na superpočítač

1. Paralelizace algoritmu
2. Hierarchické datové struktury
3. Efektivní manipulace s rozsáhlými daty



Škálovatelné (optimální) algoritmy

Numerická škálovatelnost:

cena (čas) $\approx$ počet neznámých

Paralelní škálovatelnost:

čas $\approx 1/\text{počet procesorů}$



Příklad: pracnost řešení soustav lineárních rovnic

Eliminace – počet operací $\approx n^3$

$$n = 10^8 \Rightarrow \text{pocet operací} \approx 10^{24}$$

$$\text{NUDT Tianhe} - 2 \approx 10^8 \text{ sec} \approx 1 \text{ rok}$$

Rozklad na 10000x10000 rovnic (???)

$$\Rightarrow 10000 \times 10000^3 \approx 10^{16} \text{ operací}$$

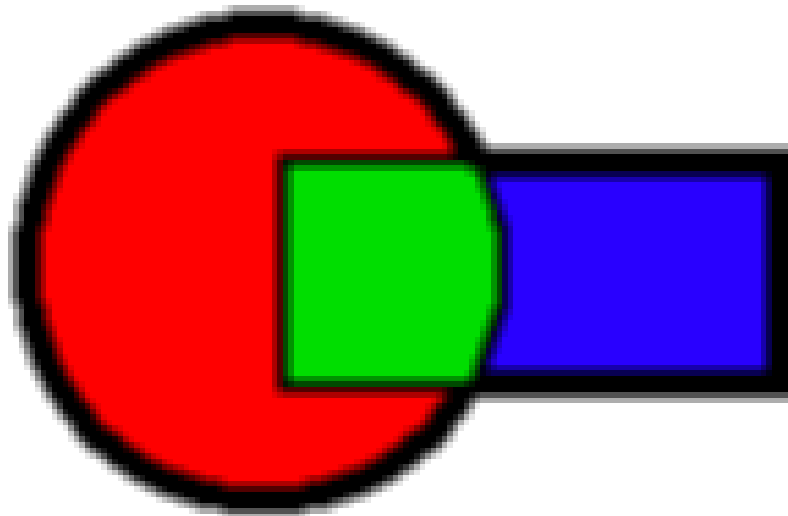
10000 procesoru

$$\Rightarrow 10^{12} \text{ operací na procesoru} \approx 0.001 \text{ sec}$$



Rozložení oblasti I.

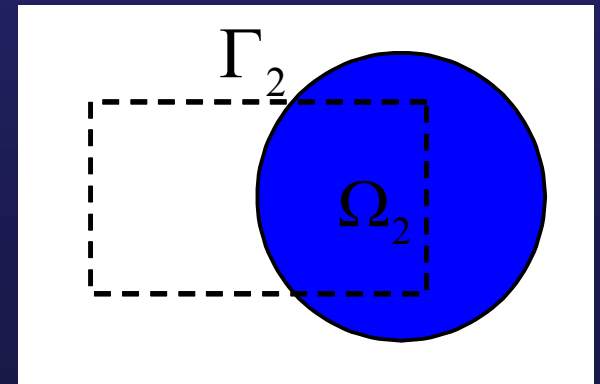
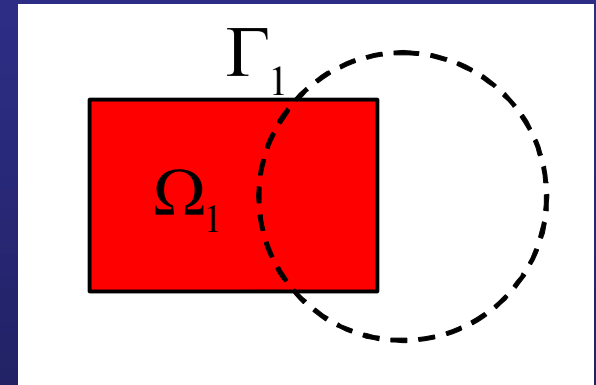
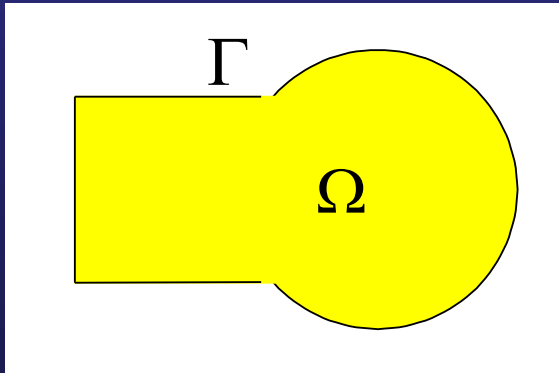
H. A. Schwarz (1869)



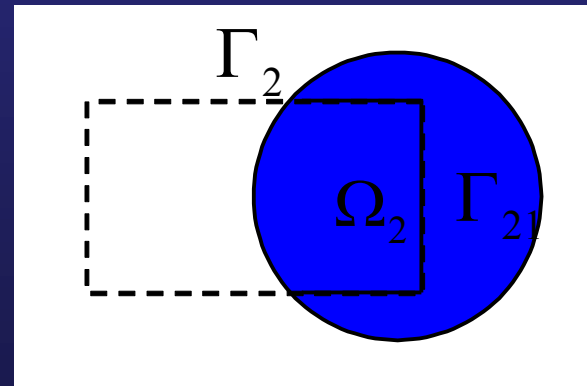
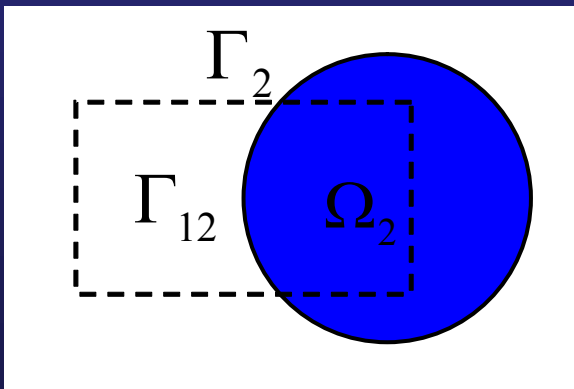
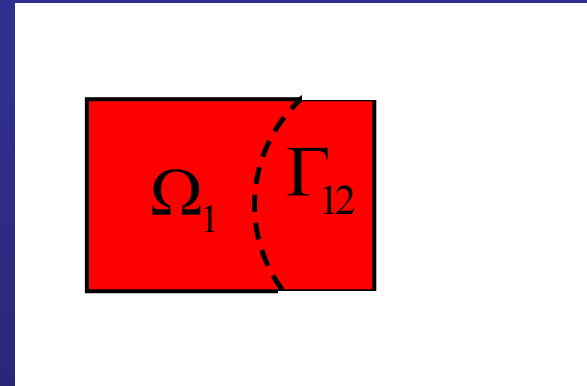
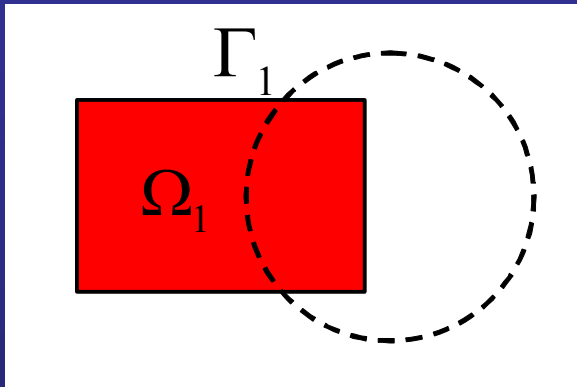
Logo ddm.org

Rozložení oblasti s překrytím

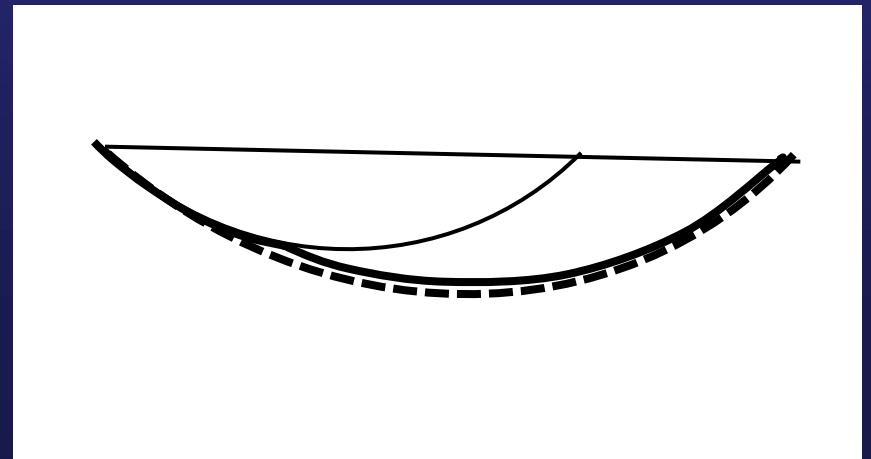
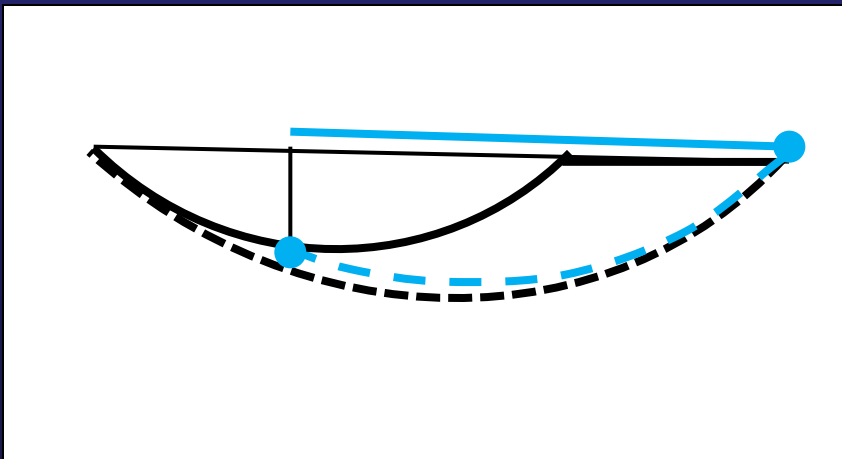
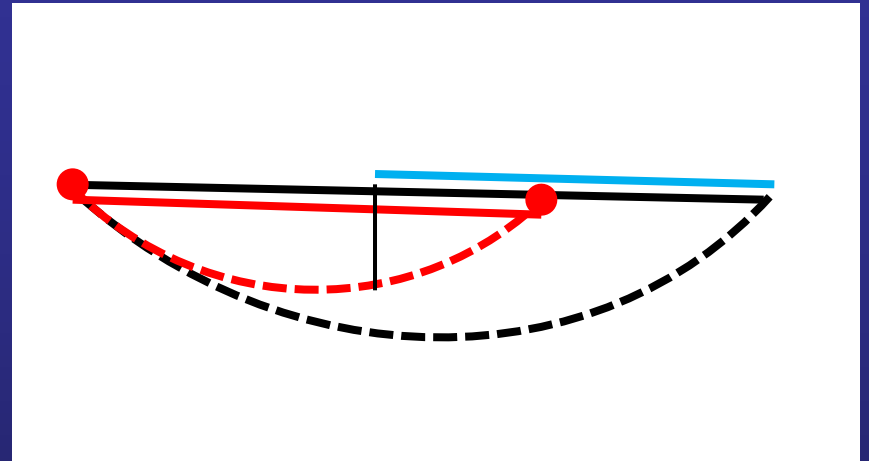
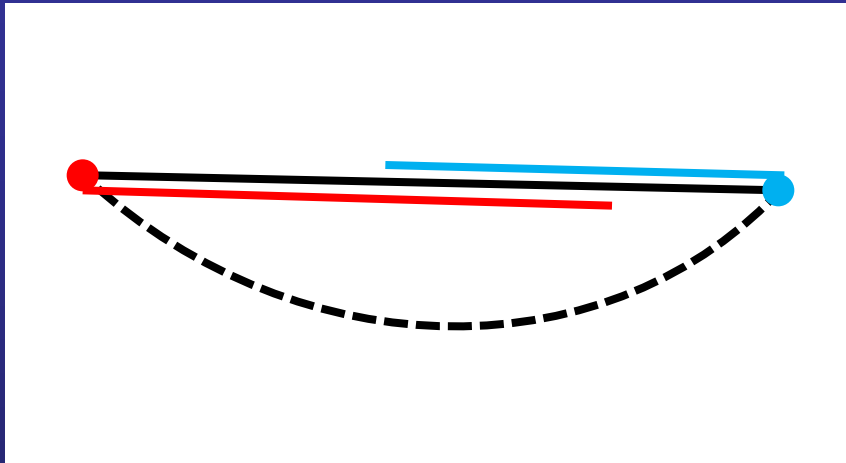
$$\begin{array}{lll} \Delta u = f & \text{v} & \Omega \\ u|_{\Gamma} = 0 & \Rightarrow & \begin{array}{ll} \Delta u_1 = f_1 & \text{v} & \Omega_1 \\ u_1|_{\Gamma_1} = 0 & & \\ u_1 = u_2 & \text{v} & \Omega_1 \cap \Omega_2 \end{array} \\ & & \begin{array}{ll} \Delta u_2 = f_2 & \text{v} & \Omega_2 \\ u_2|_{\Gamma_2} = 0 & & \end{array} \end{array}$$



Schwarz's algorithm (H. A. Schwarz, 1869)

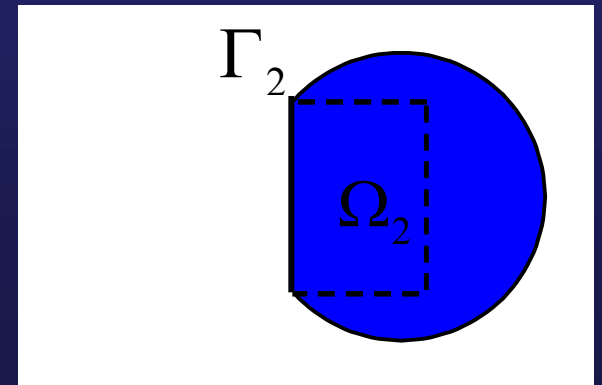
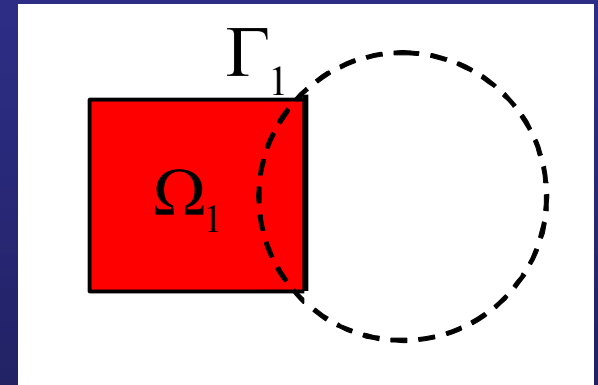
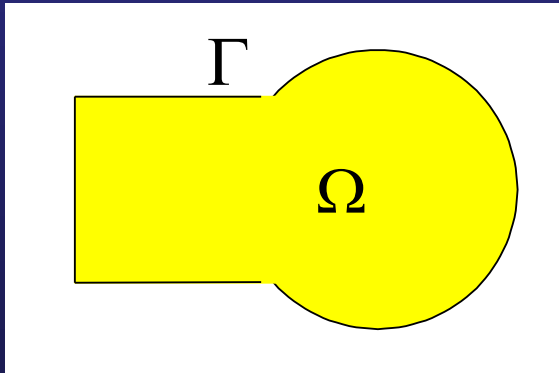


Iterace Schwarzovy metody

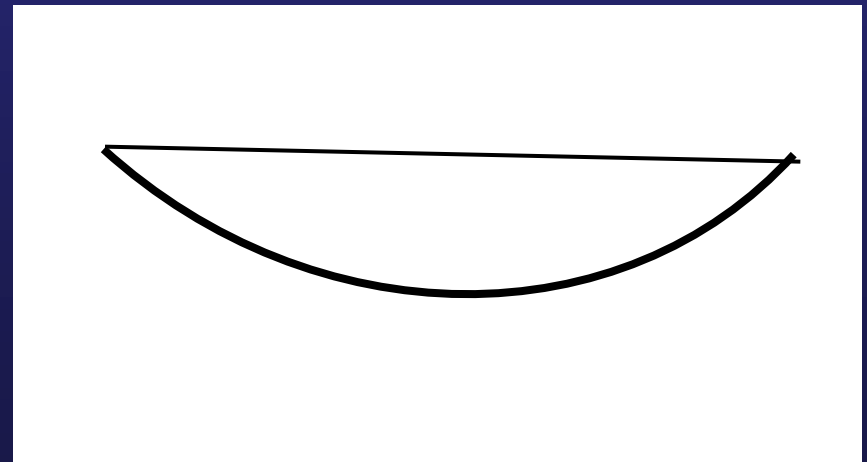
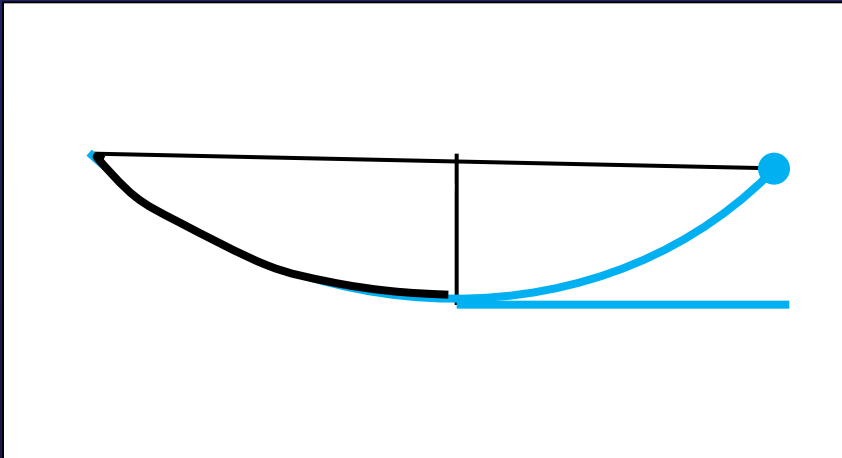
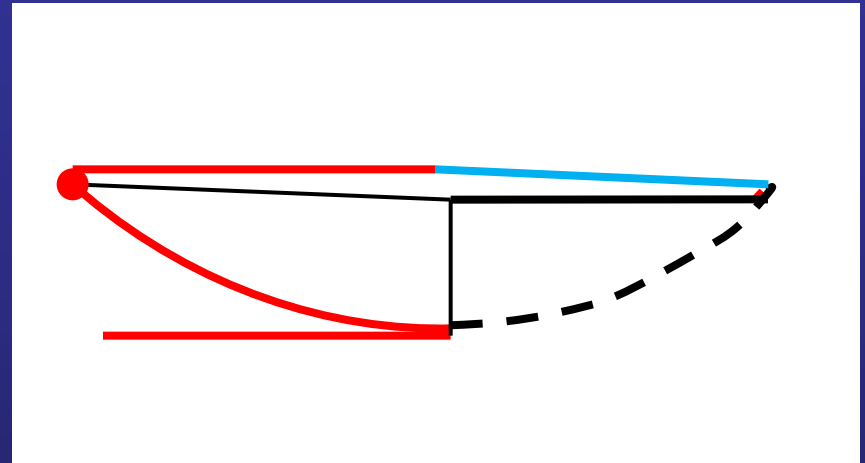
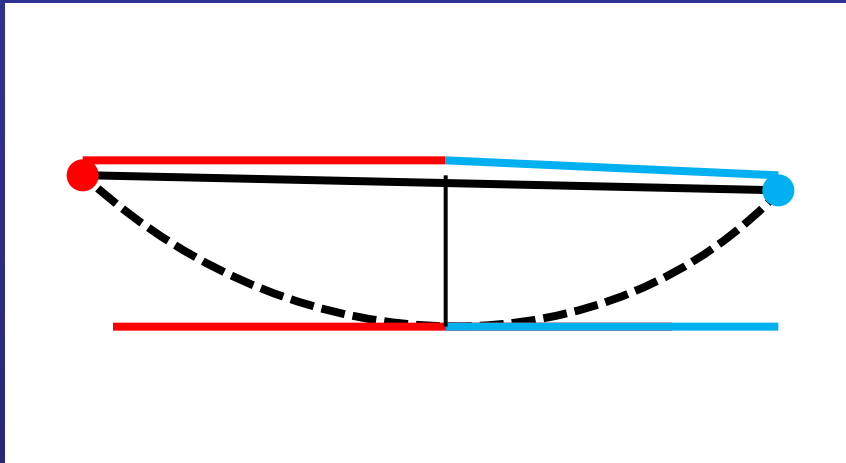


Rozložení oblasti bez překrytí-FETI

$$\begin{aligned} \Delta u &= f \quad \text{v} \quad \Omega & \Delta u_1 &= f_1 \quad \text{v} \quad \Omega_1 & \Delta u_2 &= f_2 \quad \text{v} \quad \Omega_2 \\ u|_{\Gamma} &= 0 & \Rightarrow \quad \frac{d}{dn} u_1 &= \frac{d}{dn} u_2 & \text{na} \quad \Gamma_{12} &= 0 \end{aligned}$$

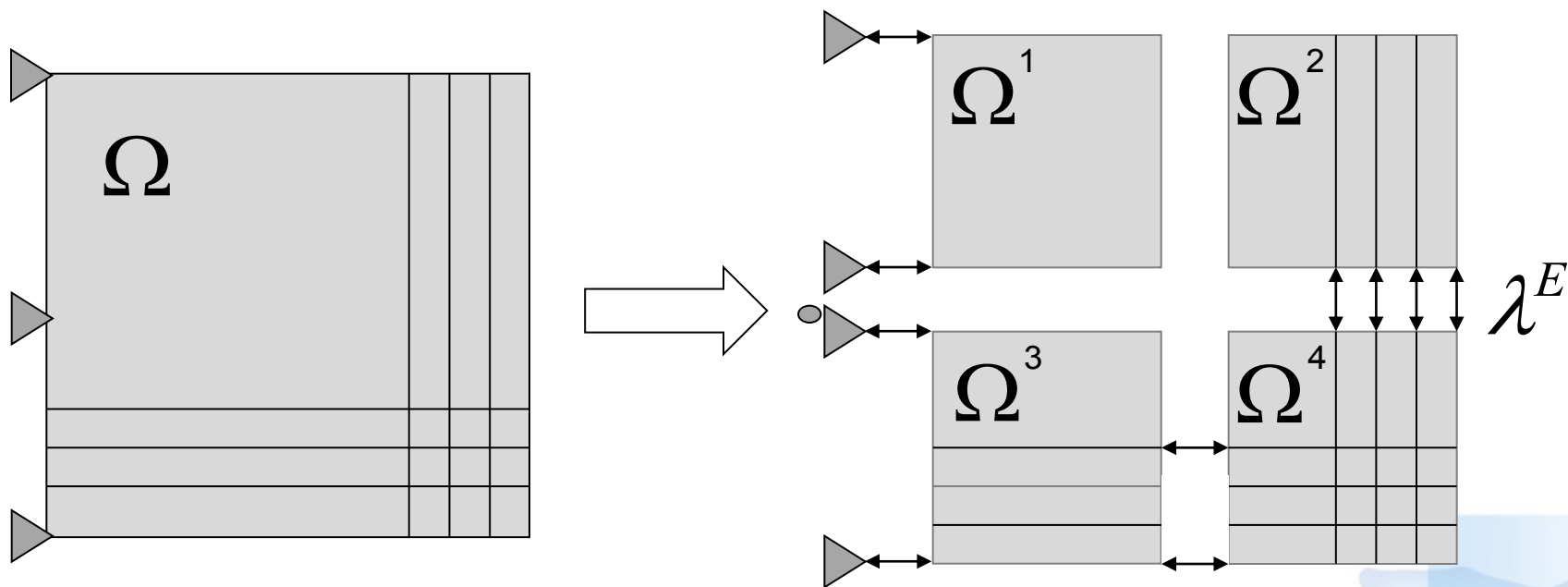


Iterace FETI metody





TFETI (AF FETI) rozložení oblasti





Řešení soustavy 300 mil. rovnic

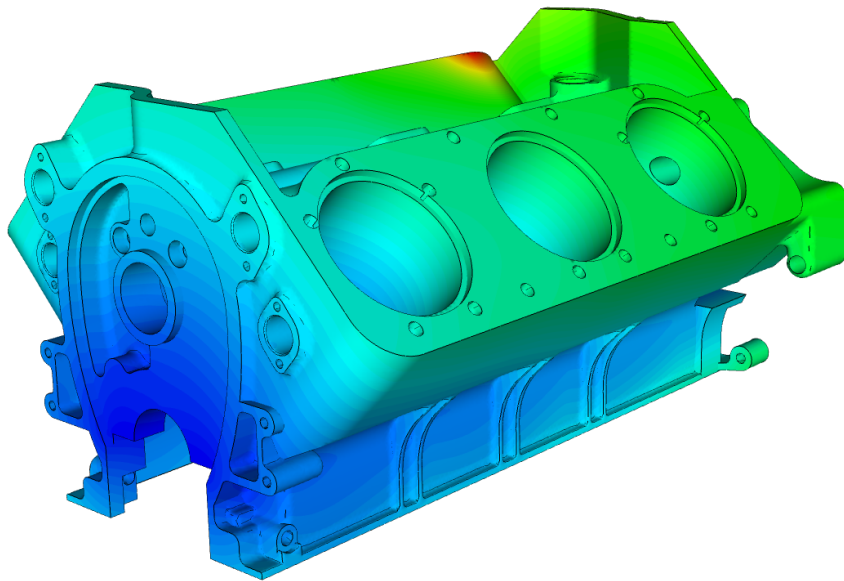
FLLOP TFETI software (V. Hapla a D. Horák)

# subdomains = # cores		1728	3072	4800
Primal variables		114 476 544	201 283 584	314 505 600
Dual variables		1 228 464	2 183 424	3 422 400
Solution time [s]	PETSc	28.90	31.70	40.51
# iterations	PETSc	42	42	42
Kernel dimension		5 184	9 216	14 400

Věta: TFETI vyřeší třídu rovnic vznikajících z daného spojitého problému za počet násobení maticí, který nezávisí na velikosti a rozložení soustavy apokud je pravidelné (Farhat, Mandel, Roux 1994)



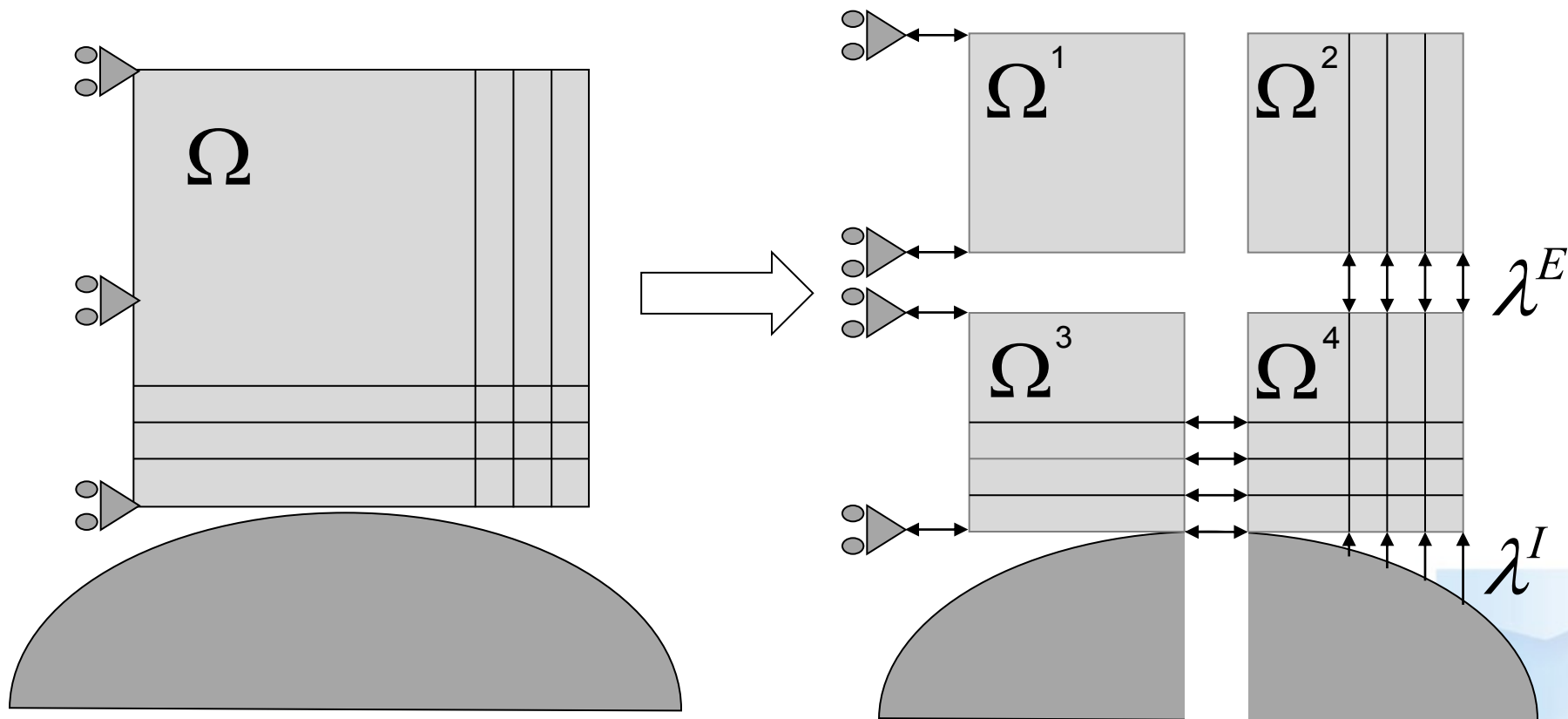
Teplotní namáhání bloku motoru 98 mil. rovnic (D. Horák, V. Hapla)



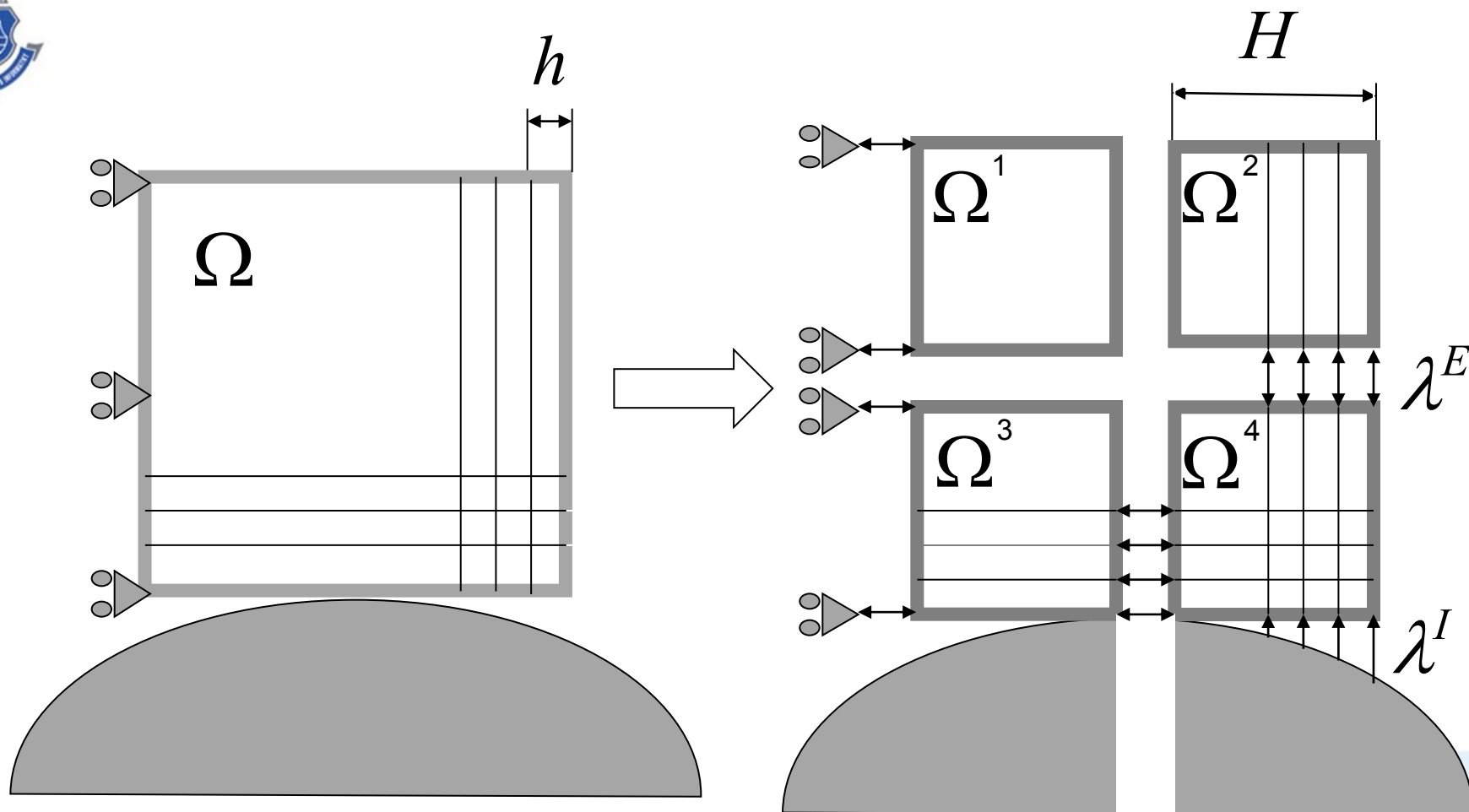
prim.dim.	98,214,558
dual.dim	13,395,882
defect	30,072
# subdomains	5,012
# iterations	181
K+ fact. time	3.9 sec
CP fact. time	18 sec
K+ act. time	15 sec
CP sol. time	75 sec
iter. sol. time	233 sec
time per 1 iter	1.28 sec
total time	~260 sec



TFETI (AF FETI) rozložení oblasti pro kontakt



TBETI (AF BETI) rozložení oblasti



Linear problems Langer and Steinbach Computing 2003
Variational inequalities Bouchala, Z.D., Sadowská Comp. 2008, 2009,
EABE 2011



Optimalita TFETI/TBETI se SMALSE/MPGP pro kontaktní úlohy (bez tření, Tresca, dynamické)

Theorem:

An approximate solution λ_i^k of the contact problem
TFETI / TBETI with n unknowns which satisfies

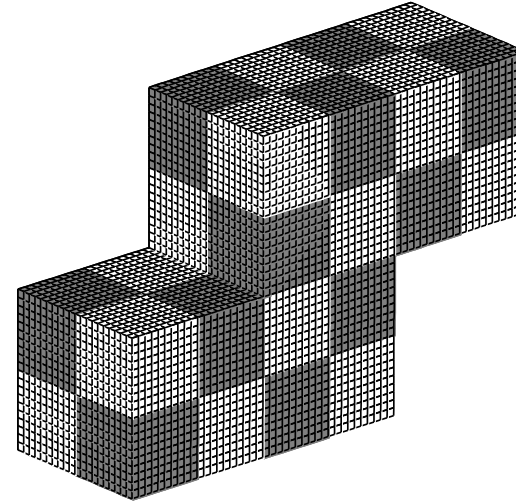
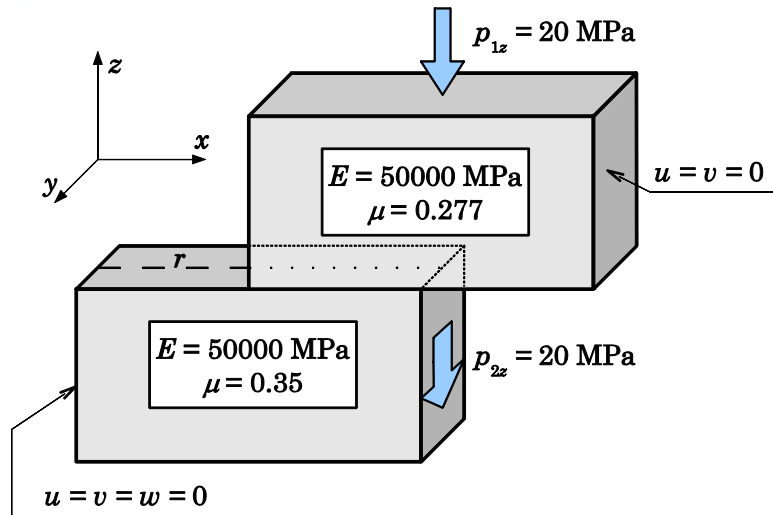
$$\left\| \lambda_i^k - \hat{\lambda}_i \right\| \leq \varepsilon \left\| \mathbf{d}_i \right\| \quad \text{and} \quad \left\| \mathbf{g}^P \left(\lambda_i^k \right) \right\| \leq \varepsilon \left\| \mathbf{d}_i \right\|$$

can be found with MPGP / SMALSE in $\mathbf{O}(1)$ iterations
at the cost proportional to n .

Z.D., Kozubek, Markopoulos, Brzobohatý, Vondrák, Horyl IJNME 2010
M.Sadowská, Z.D., Kozubek, Markopoulos, Bouchala EABE 2011,
Z.D., Kozubek, Markopoulos, Brzobohatý, horyl CMAME 2011...



Scalability of TFETI – 3D Tresca

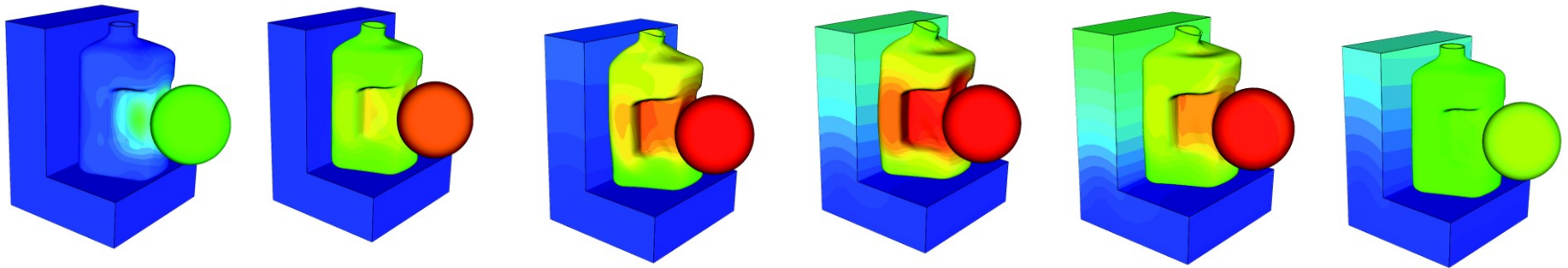


Primal dimension	Dual dimension	Subdomains	Null space	Matrix-vector	Outer iterations
431244	88601	108	648	131	3
5478396	1261493	1372	8232	146	4
11563588	2728955	2916	17496	172	4





I : Transient contact problems





Časově závislá úloha (T. Brzobohatý)

Number of Subdomains	16	54	250
Primal variables	196 608	663 552	3 072 000
Dual variables	21 706	81 652	443 920
MPRGP iterations	67	86	191
MPRGP-P iterations	60	67	112

Number of Subdomains	16	54	250
Primal variables	196 608	663 552	3 072 000
Dual variables	21 706	81 652	443 920
MPRGP iterations	39	40	42
MPRGP-P iterations	40	40	42

Závěr

1. Pořád se dá něco vymýšlet
2. Paralelní počítače vedou k novému pohledu na výpočetní metody
3. Dnes se řeší úlohy, o kterých se před 10ti lety nikomu ani nezdálo
4. To vše se dělá i na VŠB-TUO!!!
(Centrum excellence IT4I+KAM)