

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Propojení optimalizačních programů s
kvantověchemickými softwarovými balíky
*(jeho využití při zkoumání nadplochy
potenciální energie)*

Lukáš Červenka

Frýdlant nad Ostravicí 2012

STŘEDOŠKOLSKÁ ODBORNÁ ČINNOST

Obor: 18 Informatika

Propojení optimalizačních programů s kvantověchemickými softwarovými balíky

(jeho využití při zkoumání nadplochy potenciální energie)

Connection between optimization programs and quantum chemical software packages

(its use in exploring hypersurface of potential energy)

Autor:	Lukáš Červenka
Škola:	Gymnázium Frýdlant nad Ostravicí, příspě. org. nám. T. G. Masaryka 1260 739 11 Frýdlant nad Ostravicí
Konzultant:	Doc. RNDr. René Kalus, Ph.D.

Frýdlant nad Ostravicí 2012

Prohlášení

Prohlašuji, že jsem svou práci vypracoval(a) samostatně, použil(a) jsem pouze podklady (literaturu, SW atd.) uvedené v příloženém seznamu a postup při zpracování a dalším nakládání s prací je v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) v platném znění.

V dne podpis:

Poděkování

Děkuji Doc. RNDr. René Kalusovi, Ph.D. za vedení při vědecké stáži v rámci projektu Otevřená věda II. Také bych chtěl poděkovat organizátorům tohoto projektu a svému učiteli fyziky Mgr. Lukáši Bjolkovi, který mě k Otevřené vědě přivedl.

Anotace

Rozlišujeme dva druhy stacionárních bodů: extrémy (minima a maxima) a sedlové body. Ve fyzice molekul reprezentují jednotlivá minima strukturní izomery a sedlové body takové body, kterými při změně prochází jeden izomer v izomer druhý. Existuje mnoho algoritmů, které dokáží spolehlivě najít minima funkce, a méně algoritmů, které jsou schopny nalézt sedlové body. Současně existuje několik kvantověchemických softwarových balíčků, které jsou schopny velmi přesně numericky vypočítat potenciální energii určité konfigurace molekuly, gradient této energie apod. Úkolem mé práce bylo propojit programy implementující optimalizační algoritmy s kvantověchemickými programy, zajistit vzájemnou výměnu dat, výpočet paralelizovat, zaznamenávat průběh výpočtu a ošetřit výjimky běhu. Také bylo třeba zachovat naprostou obecnost řešení, je tedy možno libovolný program vyjmout a nahradit programem jiného dodavatele při zachování funkčnosti. Bylo rovněž vytvořeno několik vlastních optimalizačních programů (horolezecký algoritmus, basin-hopping Monte Carlo a evoluční strategie). Vyvinuté programy byly použity k lokální optimalizaci molekul dusíku, vody, amoniaku a k hledání strukturních izomerů iontových klastrů vzácných plynů Ar_3^+ , Ar_4^+ , He_3^+ a He_4^+ .

Klíčová slova

propojení optimalizačních programů s kvantověchemickými softwarovými balíky, simulace, potenciální energie, paralelizace výpočtu, kvantověchemické softwarové balíky, iontové klastry vzácných plynů, basin hopping, evoluční strategie, helium, argon

Obsah

Prohlášení	3
Poděkování.....	4
Anotace.....	5
Klíčová slova.....	5
Úvod.....	7
Metody.....	8
Propojení simulačních a interakčních programů.....	8
Způsob propojení.....	8
Diagram propojení.....	8
Jednotlivé programy.....	9
Řídicí program – ctr.sh.....	9
Simulační program – simul.....	9
Interakční program – kvantověchemický softwarový balík.....	9
Interpretační programy – simul_int_input, simul_int_output, inter_int_... ..	9
Definice formátu XYZ.....	10
Paralelizace.....	11
Dvojitá paralelizace.....	11
Technické provedení.....	11
Spouštění, běh a vyhodnocení výpočtu.....	11
Vlastní simulační programy.....	11
Simul v1 a v2 – horolezecký algoritmus.....	12
Popis horolezeckého algoritmu.....	12
Simul v3 – Basin hopping.....	12
Popis Basin hopping algoritmu.....	12
Simul v4 – evoluční strategie.....	13
Popis evolučních strategií.....	13
Výsledky výpočtů.....	14
Výsledky dosažené pomocí horolezeckého algoritmu.....	14
N2.....	14
H20.....	14
NH3.....	15
Výsledky dosažené pomocí Basin hopping.....	16
Ar3+	16
Ar4+.....	17
He3+.....	18
He4+.....	19
Závěr.....	20
Shrnutí.....	20
Výhledy co budoucná.....	20
Použitá literatura a software.....	20
Seznam příloh.....	21

Úvod

V současné chvíli existuje mnoho efektivních optimalizačních algoritmů, které dovedou rychle najít stacionární body funkcí, a také kvalitní kvantově chemické softwarové balíky, které dokáží velmi přesně vypočítat potenciální energii molekuly. Mým cílem je propojit kvantověchemické softwarové balíky s programy implementujícími optimalizační algoritmy a zajistit jejich vzájemnou komunikaci nezávisle na jejich dodavateli.

Jedním z využití tohoto propojení je zkoumání nadplochy potenciální energie molekuly. Lokální minima potenciální energie, která lze nalézt stejně jako sedlové body pomocí optimalizačních programů, odpovídají strukturním izomerům. Sedlové body prvního řádu značí body, kterými přechází jeden izomer v druhý.

Základem propojení je řídicí program, který přepíná program simulační a interakční, zajišťuje přípravu vstupních souborů a ošetřuje výjimky. Dále existují programy interpretační, které zajišťují konverzi formátů souborů. Užitečnou vlastností propojení je více úrovní paralelizace – paralelizace výpočtu více funkčních hodnot v jednom čase a také paralelizace v rámci výpočtu jedné funkční hodnoty.

V první části práce je popsána metodika propojení, jeho využití, v části druhé jsou uveřejněny výsledky pilotních výpočtů stabilních izomerů molekul helia, argonu, vody, amoniaku a dusíku.

Metody

Propojení simulačních a interakčních programů

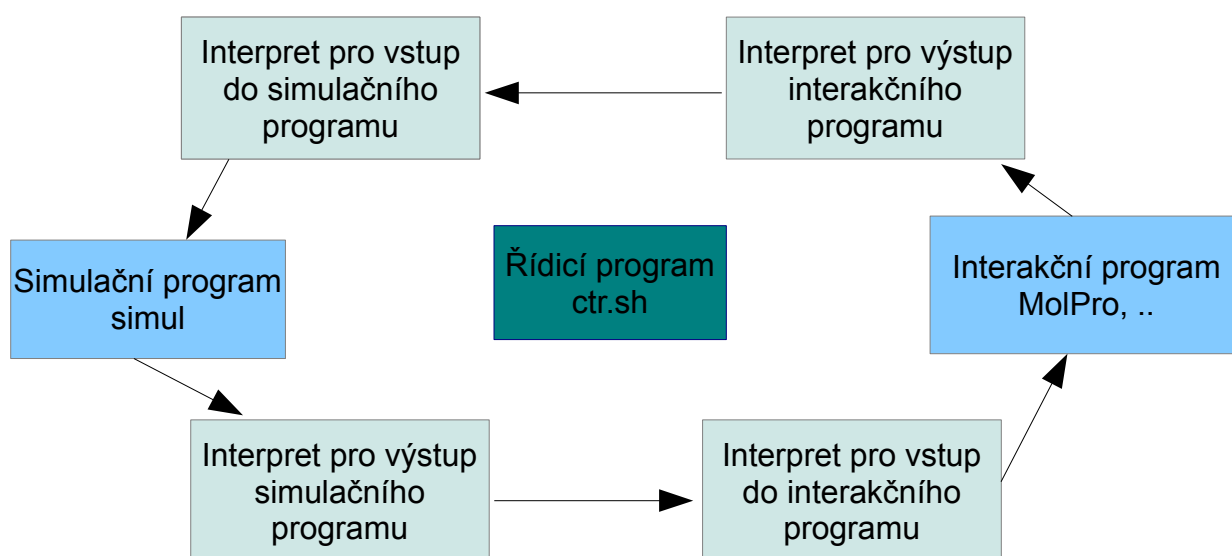
Způsob propojení

V průběhu výpočtu je potřeba střídavě spouštět simulační a interakční program, zajistit správný formát dat vstupu a výstupu obou programů, zachytávat a ošetřovat výjimky a zastavovat běh při splnění kritérií k ukončení výpočtu. Cílem bylo také vytvořit řešení obecné, nezávislé na dodavateli jednotlivých programů.

Další výhodou je možnost paralelizace výpočtu, tedy možnost počítat více funkčních hodnot zároveň. V současné době umějí paralelizovat výpočet i samotné kvantověchemické softwarové balíky, ale jedná se pouze o paralelizaci výpočtu jediné funkční hodnoty a například u mnou používaného interakčního programu Molpro bylo efektivní použít nejvýše 16 jader pro jeden výpočet. Máme-li ale procesorů více, je řídicí program schopen spustit více instancí Molpro v jeden čas – máme-li 64 jader, lze celkový výpočet urychlit při použití 16 jader na jeden výpočet čtyřikrát. Dostupných jader pro výpočet ale mohou být na výpočetních serverech až tisíce.

Uživatel před začátkem výpočtu zadá do jediného souboru všechna potřebná kritéria pro výpočet potenciální energie, simulaci, paralelizaci a stop kritéria. Každý z programů čte jen svou vyhrazenou část tohoto souboru. Průběh výpočtu je také zaznamenáván do přehledných logů.

Diagram propojení



Jednotlivé programy

Celý systém se skládá ze sedmi programů: řídicí program, interpretační programy, interakční a simulační program. Tyto programy spolu komunikují především pomocí malých souborů.

Řídicí program – *ctr.sh*

Tento program zajišťuje inicializaci výpočtu, samotné přepínání jednotlivých programů, kontrolu jejich běhu (vstupu, výstupu, návratové hodnoty) a zastavení výpočtu.

Základem je smyčka spouštějící postupně:

1. interpret vstupu simulačního programu
2. simulační program
3. interpret výstupu simulačního programu + logger
4. interpret vstupu interakčního programu
5. interakční program
6. interpret výstupu interakčního programu

Každý program je spuštěn, po jeho ukončení jsou (je-li potřeba) zkontrolovány jeho výstupní soubory a je spuštěn program další.

Simulační program – *simul*

V tomto programu probíhá samotná simulace.

Jeho vstupem jsou funkční hodnoty v bodech, o které si zažádal v předchozím kroku výpočtu. Výstupem je několik bodů, ve kterých potřebuje znát funkční hodnotu.

Interakční program – *kvantověchemický softwarový balík*

Tento program dokáže vypočítat funkční hodnotu v zadaném bodě, případně provést lokální optimalizaci, vypočítat gradient v bodě apod. Množství funkcí interaktu závisí na dodavateli daného balíku, přičemž každý optimalizační algoritmus má různé nároky na schopnosti interaktu.

Vstupem je bod, ve kterém chceme provést jednu z výše uvedených operací (např. vypočtení potenciální energie určité konfigurace). Výstupem je pak požadovaná informace.

Interpretační programy – *simul_int_input, simul_int_output, inter_int ...*

Jelikož každý simulační i interakční program používá vlastní formát vstupních a výstupních

souborů, je třeba zajistit konverzi těchto formátů. K tomu slouží dvojice interpretů přiřazená každému programu. Aby byla dodržena modulárnost, slouží pro vzájemnou komunikaci interpretů jednotlivých programů přesně definovaný formát XYZ.

Interprety bychom mohli rozdělit do dvou skupin:

1. **vstupní interprety** – zpracovávají soubor v obecném formátu a převádí jej na vstup programu v požadovaném formátu
2. **výstupní interprety** – zpracovávají výstupní soubor programu ve specifickém formátu a vytvářejí soubor v obecném formátu

Definice formátu XYZ

Jedná se o textový soubor s příponou xyz, který obsahuje informace o počtu atomů v molekule, velikosti potenciální, kinetické a celkové energie, poloze jednotlivých atomů, chemických prvcích a číslu iterace.

POČET_ATOMŮ

ČÍSLO_ITERACE CELKOVÁ_ENG E_p = POTENCIÁLNÍ_ENG E_k = KINETICKÁ_ENG

PRVEK1 SOUŘADNICE_X1 SOUŘADNICE_Y1 SOUŘADNICE_Z1

PRVEK2 SOUŘADNICE_X2 SOUŘADNICE_Y2 SOUŘADNICE_Z2

...

PRVEKn SOUŘADNICE_Xn SOUŘADNICE_Yn SOUŘADNICE_Zn

Příklad pro Ar₄⁺:

4

15 -2106.6738430 E_p = -2106.6738430 E_k = 0

Ar 2.9460293 -1.3360406 0.0002558

Ar -0.0140175 2.8235344 -0.0003156

Ar -0.2496813 -1.1089179 -0.0001665

Ar -2.7510928 -0.8783919 0.0002195

Paralelizace

Dvojí paralelizace

Interpretační programy obvykle umějí paralelizovat právě probíhající výpočet. Každá paralelizace má ale jistou režii a nad určitý počet procesorů na jeden výpočet tato režie stoupne nad mez výhodnosti paralelizace. Z tohoto důvodu je použita paralelizace dvojí – spustí se několik instancí interaktu v jeden okamžik, přičemž každá instance rovněž paralelizuje. Pokud tedy máme stroj s 320 procesory a jedna instance optimálně využije maximálně 16 procesorů, výpočet se urychlí až dvacetkrát.

Využití pro dvojí paralelizaci nabízí jen některé optimalizační algoritmy, například evoluční strategie nebo horolezecký algoritmus. V obou případech každá instance interaktu provádí výpočet právě jednoho jedince.

Technické provedení

Na začátku se rozdělí vstupní soubor pro interpret v univerzálním formátu na několik částí. Tyto jednotlivé části se předají interpretu pro vstup do interakčního programu. Každé instanci se vytvoří pracovní adresář, do kterého se tyto soubory překopírují a spustí se první dávka instancí interaktu. Ihned po doběhnutí jedné instance se spustí instance další. Po dokončení všech výpočtů se jednotlivé výstupy interakčního programu spojí ve vhodném pořadí do jednoho souboru a předají interpretu pro vstup do simulačního programu.

Spouštění, běh a vyhodnocení výpočtu

Pro nastavení všech parametrů programů a informací potřebných pro výpočet slouží soubor `start.ini`, který je členěný do čtyř sekcí.

1. Simulační – nastavení parametrů simulace
2. Interakční – nastavení kvantověchemické báze, metody výpočtu energie
3. Popis molekuly – souřadnice atomů počáteční molekuly, celkový náboj, typ atomů
4. Popis spouštění interaktu – nastavení počtu procesorů pro jednu instanci, pro paralelizaci

Průběh výpočtu se zaznamenává do logů pomocí interpretu výstupu simulačního programu. Při lokální optimalizaci se zaznamenává vývoj energie v průběhu výpočtu, v případě basin-hopping algoritmu nalezená minima.

Vlastní simulační programy

V rámci usazení protokolu propojení jsem pracoval s dvěma typy optimalizačních algoritmů.

První typ vyžadoval výpočet energie určité konfigurace, druhý typ potřeboval vypočítat gradient respektive lokálně optimalizovat molekulu v každém kroku výpočtu.

Simul v1 a v2 – horolezecký algoritmus

V prvních dvou verzích simulačního programu byl použit horolezecký algoritmus. První verze programu byla schopna lokálně optimalizovat funkci jedné neznámé, tedy nalézt stabilní konfiguraci dvojatomové molekuly. Druhá verze už uměla najít minimum funkce s velkým počtem neznámých, optimalizovala n-atomové molekuly.

Popis horolezeckého algoritmu

Jedná se o algoritmus jednoduchý, robustní, ovšem poměrně náročný na množství výpočtů. Slouží k lokální optimalizaci funkce.

Na začátku výpočtu máme konfiguraci molekuly specifikovanou v souboru start.ini. Program v každém kroku vygeneruje několik pozměněných konfigurací tak, že s každým atomem pohne o náhodou vzálenost limitovanou shora. Všechny konfigurace z daného kroku se předají interaktu, který vypočte hodnotu potenciální energie každé konfigurace, přičemž z konfigurace s nejnižší energií se generuje další skupina konfigurací. Pokud se konfigurace velmi přiblíží minimu, nastane případ, že všechny nově vygenerované konfigurace mají vyšší energii než konfigurace, z které bylo generováno. V tomto případě se zmenší limit vzdálenosti posunutí atomu a pokračuje se opětovným generováním ze stejné konfigurace, zároveň se také navýší počet vytvářených konfigurací. Výpočet končí v případě, že rozdíl mezi starou konfigurací a novými konfiguracemi klesne pod stanovenou mez, nebo v případě, že je překročen maximální počet kroků.

Simul v3 – Basin hopping

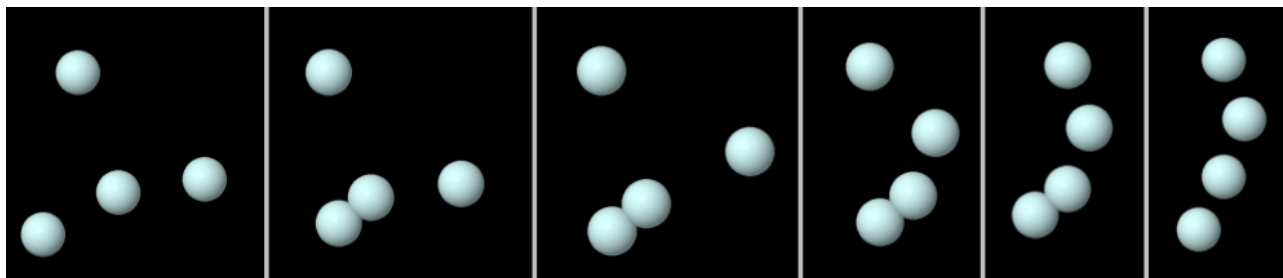
V této verzi simulačního programu je implementován algoritmus Basin hopping neboli Monte Carlo optimalizace. Program je tak určený k hledání strukturních izomerů molekuly.

Popis Basin hopping algoritmu

Výpočet začíná lokální optimalizací konfigurace uvedené ve start.ini, kterou provede samotný kvantověchemický balík. Simulační program obdrží energii prvního minima, kterou zaznamená. V Vdalších krocích vždy vezme poslední přijatou konfiguraci, pozmění v ní náhodně polohu jednoho atomu (maximální posun je omezen shora, stejně jako maximální vzdálenost atomu od těžiště molekuly) a tuto konfiguraci nechá opět lokálně optimalizovat. Pokud se nalezené minimum neliší od minulého, nebo má ještě nižší energii, přijme se ta konfigurace, z které se do něj došlo. Není-li tomu tak, stále existuje jistá pravděpodobnost, že konfigurace bude přijata. V průběhu výpočtu se

tedy nezaznamenávají jen přijatá minima, ale především všechna nalezená minima včetně nepřijatých.

Průběh změny trojúhelníkové konfigurace Ar_4^+ na lineární



Simul v4 – evoluční strategie

Simulační program čtvrté verze bude sloužit k hledání strukturních izomerů molekul, v současné době je ještě vyvíjen.

Evoluční strategie patří do skupiny algoritmů, které se inspirovaly přírodou, především přirozeným výběrem. Každého žijícího jedince si můžeme představit jako uspořádanou n -tici hodnot neznámých funkce, která svou hodnotou vyjadřuje jeho schopnost přežít v konkurenci ostatních jedinců. Každý jedinec je rozdílný, hodnoty neznámých se liší a také se liší jeho schopnost přežít. Evoluční strategie pracují na principu křížení jedinců mezi sebou a jejich náhodné mutace.

Popis evolučních strategií

Na začátku výpočtu je vygenerována populace několika jedinců. Tyto jedince vzájemně zkřížíme. Existuje mnoho způsobů křížení, implementovány jsou dva typy:

1. křížení průměrem – hodnoty dané neznámé rodičů se sečtou a vydělí počtem rodičů
2. proužkové křížení – střídají se rodiče k poskytnutí hodnoty dané proměnné

Následně jedince necháme mutovat, v tomto kroku je náhodně vybraná hodnota, která je náhodně změněna. Poté se nechá vypočíst energie každého jedince, jedinci se setřídí podle energie a ti nejlepší se použijí ke generování nových jedinců.

Za jedince si představme konfigurace molekuly a za fitness funkci (energii udávající schopnost přežít) potenciální energii konfigurace. Pomocí tohoto algoritmu tak dokážeme nalézt globální minimum funkce (nejlepšího jedince).

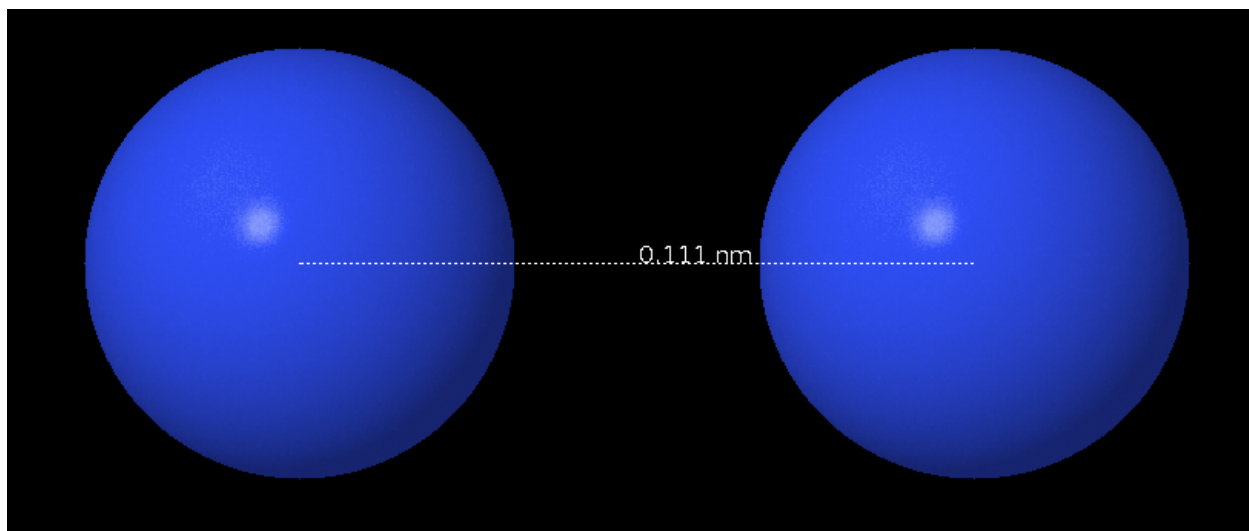
Výsledky výpočtů

Výsledky dosažené pomocí horolezeckého algoritmu

Výpočet byl spuštěn pro tři metody výpočtu energie a dvě kvantověchemické báze. Celkem tedy bylo provedeno šest lokálních optimalizací, nalezené struktury byly porovnány s konfiguracemi, které našel program MolPro. Účelem bylo především vyzkoušet systém propojení.

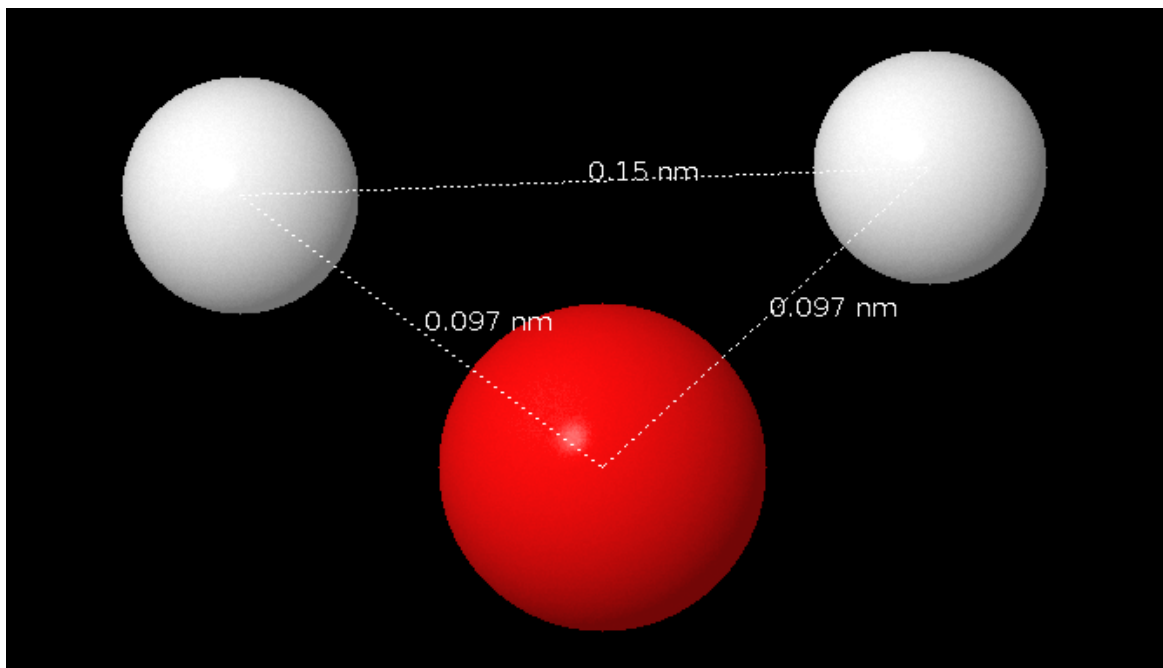
N₂

Metoda	Báze	Odchylka vzdáleností	Odchylka energie
hf	cc-pVDZ	$1,0 \cdot 10^{-4}$ ang	0mEv
hf	cc-pVTZ	$2,0 \cdot 10^{-5}$ ang	0mEv
ks,b97r	cc-pVDZ	$4,2 \cdot 10^{-5}$ ang	0mEv
ks,b97r	cc-pVTZ	$2,2 \cdot 10^{-4}$ ang	$5,4 \cdot 10^{-3}$ mEv
hf,mp2	cc-pVDZ	$3,0 \cdot 10^{-5}$ ang	0mEv
hf,mp2	cc-pVTZ	$1,3 \cdot 10^{-5}$ ang	0mEv

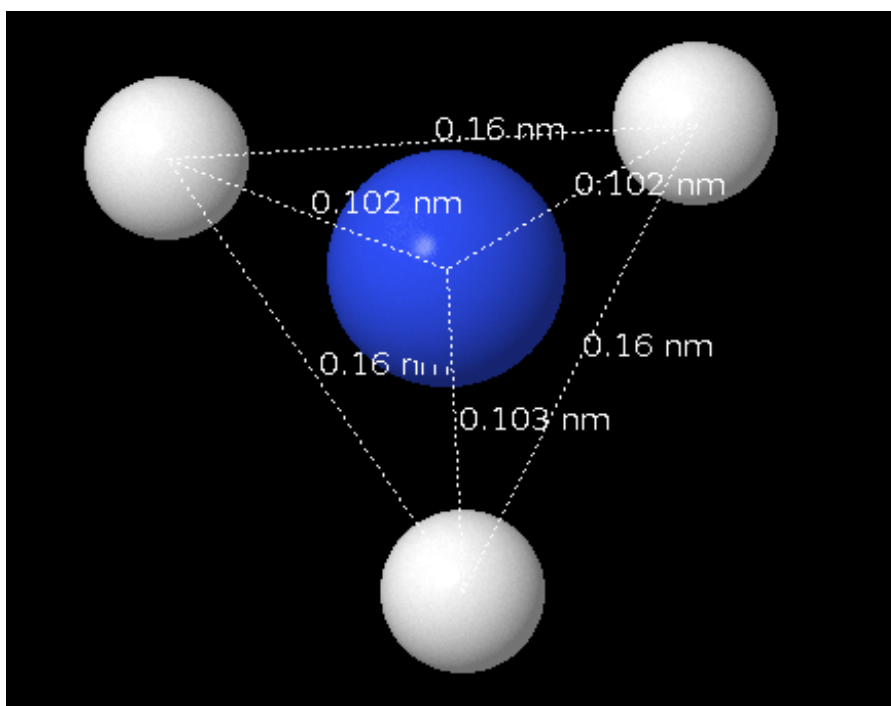


H₂O

Metoda	Báze	Odchylka vzdáleností	Odchylka energie
hf	cc-pVDZ	$7,6 \cdot 10^{-4}$ ang	$1,9 \cdot 10^{-2}$ mEv
hf	cc-pVTZ	$1,7 \cdot 10^{-4}$ ang	$2,4 \cdot 10^{-2}$ mEv
ks,b97r	cc-pVDZ	$3,1 \cdot 10^{-4}$ ang	$8,2 \cdot 10^{-2}$ mEv
ks,b97r	cc-pVTZ	$2,1 \cdot 10^{-4}$ ang	$3,1 \cdot 10^{-2}$ mEv
hf,mp2	cc-pVDZ	$6,7 \cdot 10^{-4}$ ang	$3,5 \cdot 10^{-2}$ mEv
hf,mp2	cc-pVTZ	$4,6 \cdot 10^{-4}$ ang	$1,2 \cdot 10^{-1}$ mEv



NH₃



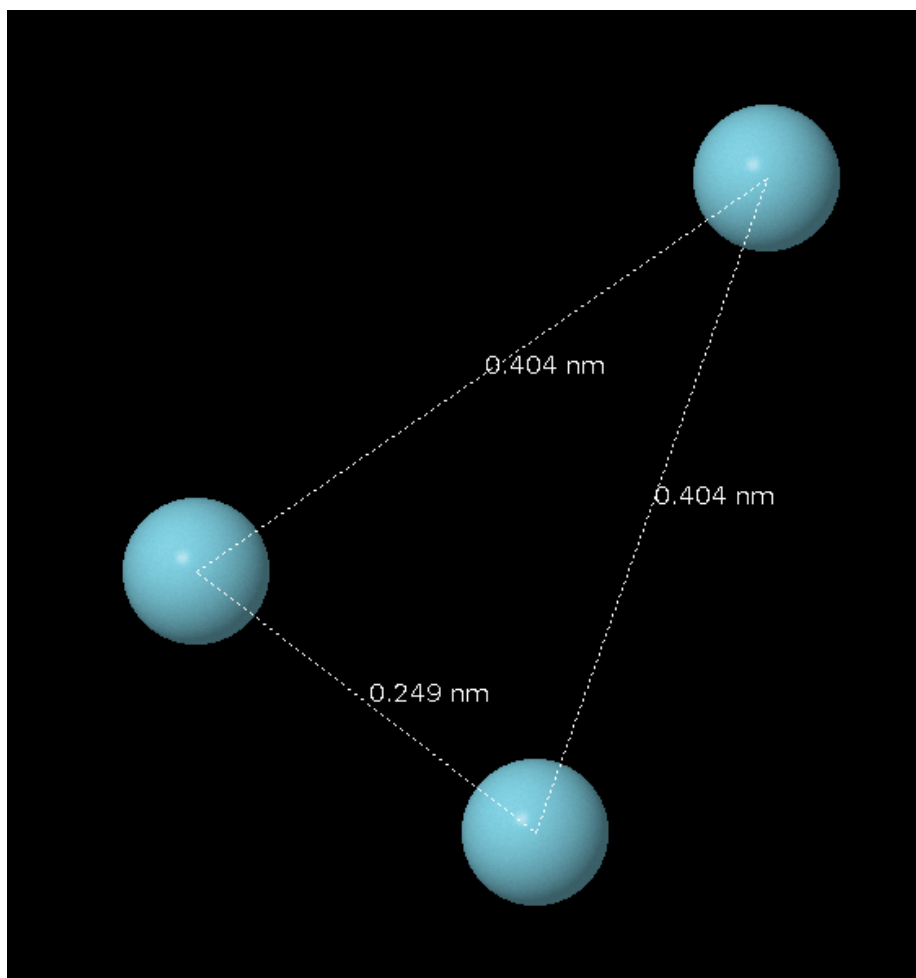
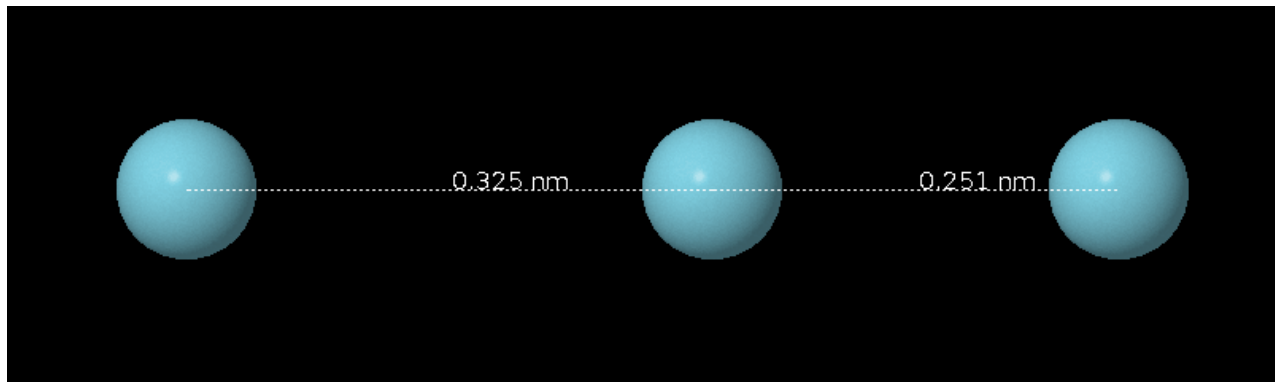
Metoda	Báze	Odchylka vzdáleností	Odchylka energie
hf	cc-pVDZ	$1,0 \cdot 10^{-4}$ ang	7,8 mEv
hf	cc-pVTZ	$3,7 \cdot 10^{-3}$ ang	$9,2 \cdot 10^{-1}$ mEv
ks,b97r	cc-pVDZ	$3,3 \cdot 10^{-3}$ ang	$6,7 \cdot 10^{-1}$ mEv
ks,b97r	cc-pVTZ	$4,4 \cdot 10^{-3}$ ang	1,2 mEv
hf;mp2	cc-pVDZ	$1,3 \cdot 10^{-3}$ ang	1,3 mEv
hf;mp2	cc-pVTZ	$1,5 \cdot 10^{-4}$ ang	2,8 mEv

Výsledky dosažené pomocí Basin hopping

Strukturální izomery argonu byly počítány z důvodu ověření přesnosti výsledků, protože jsou dobře známy referenční modely. Naopak pro helium známá spolehlivá data nejsou k dispozici. V literatuře se sice objevují výsledky výpočtů, zásadně se ale liší.

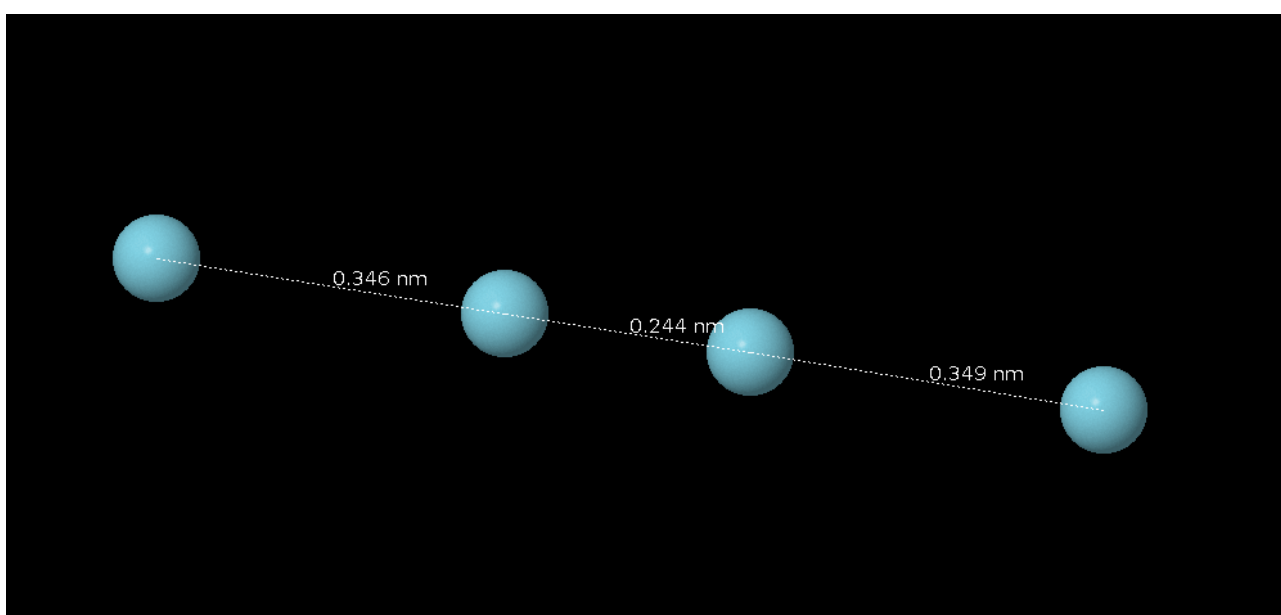
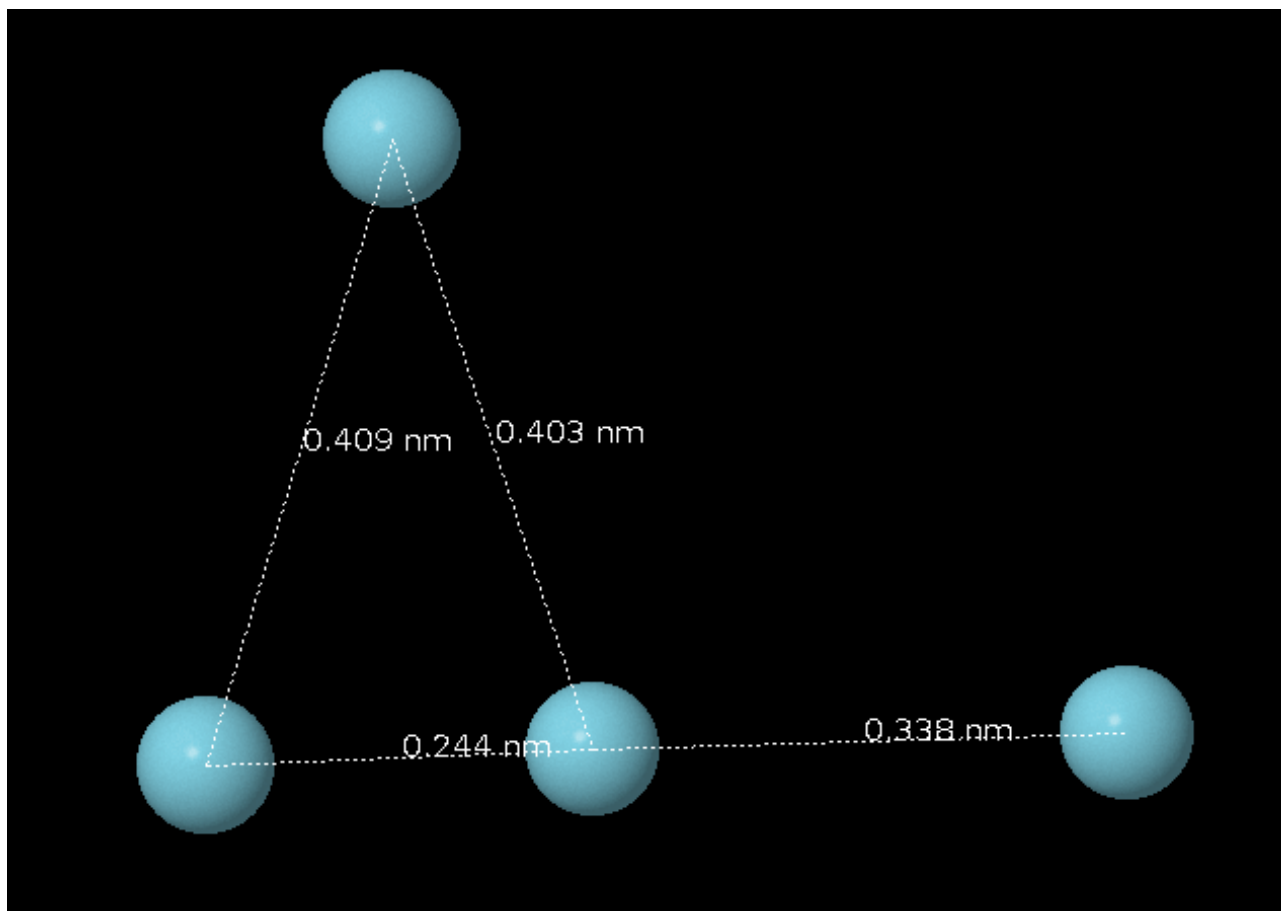
Ar₃⁺

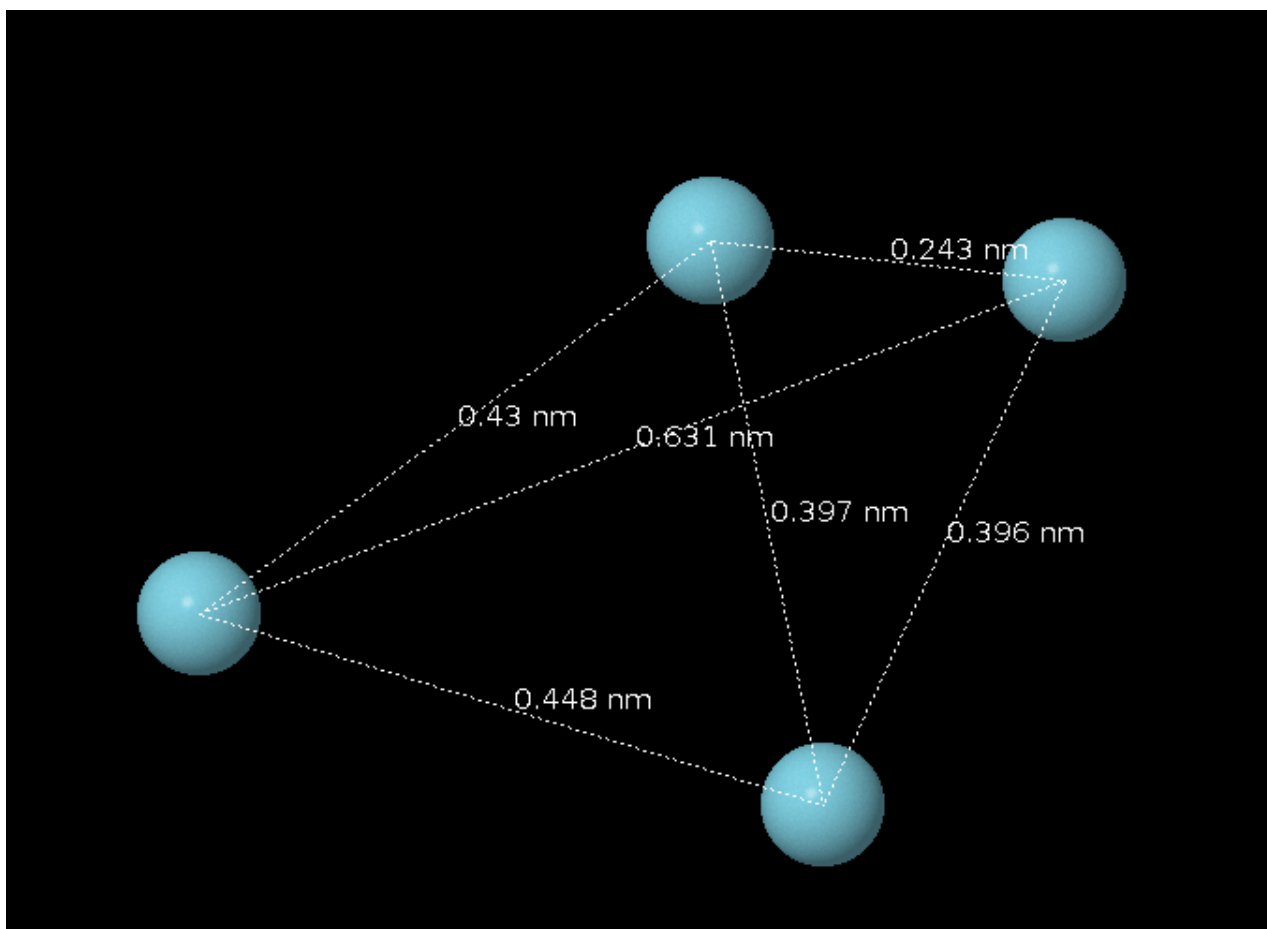
Pro výpočet byla použita kvantověchemická metoda Hartree-Fock, báze double-zeta. Byly nalezeny dva izomery.



Ar₄⁺

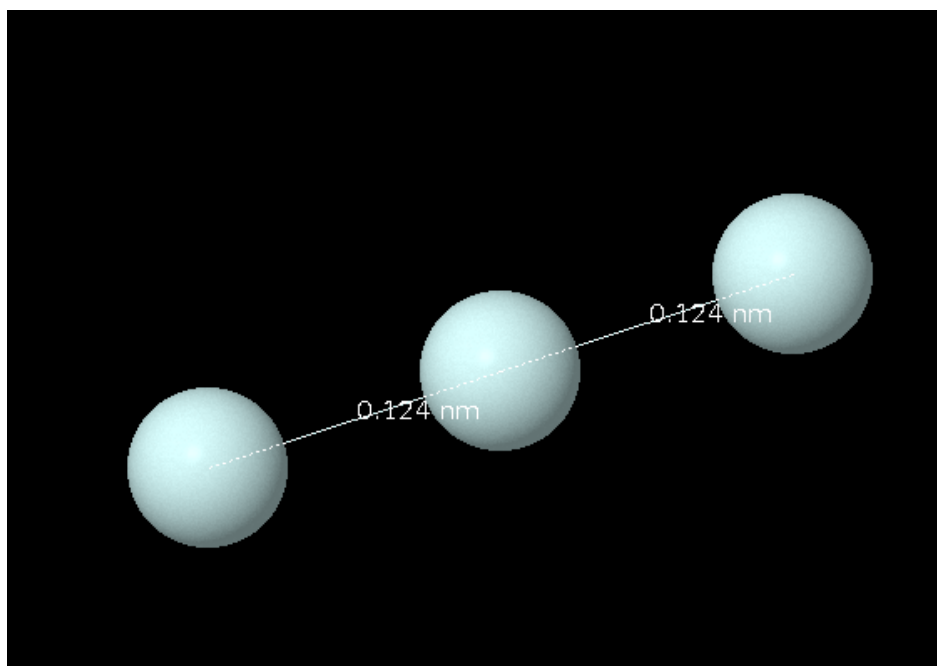
Pro výpočet byla použita kvantověchemická metoda Hartree-Fock, báze triple-zeta. Byly nalezeny tři izomery.





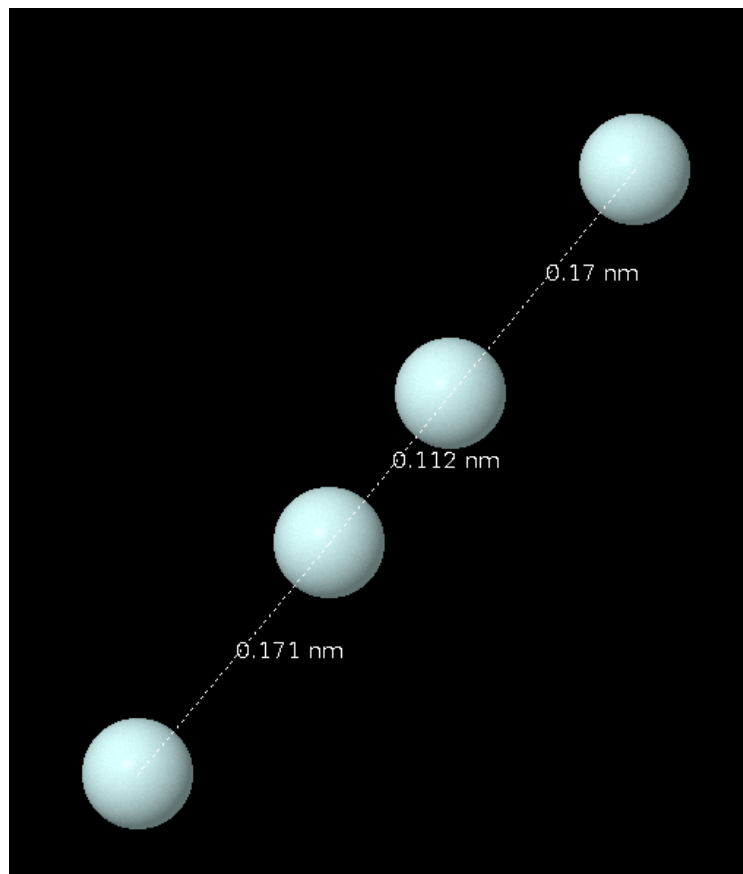
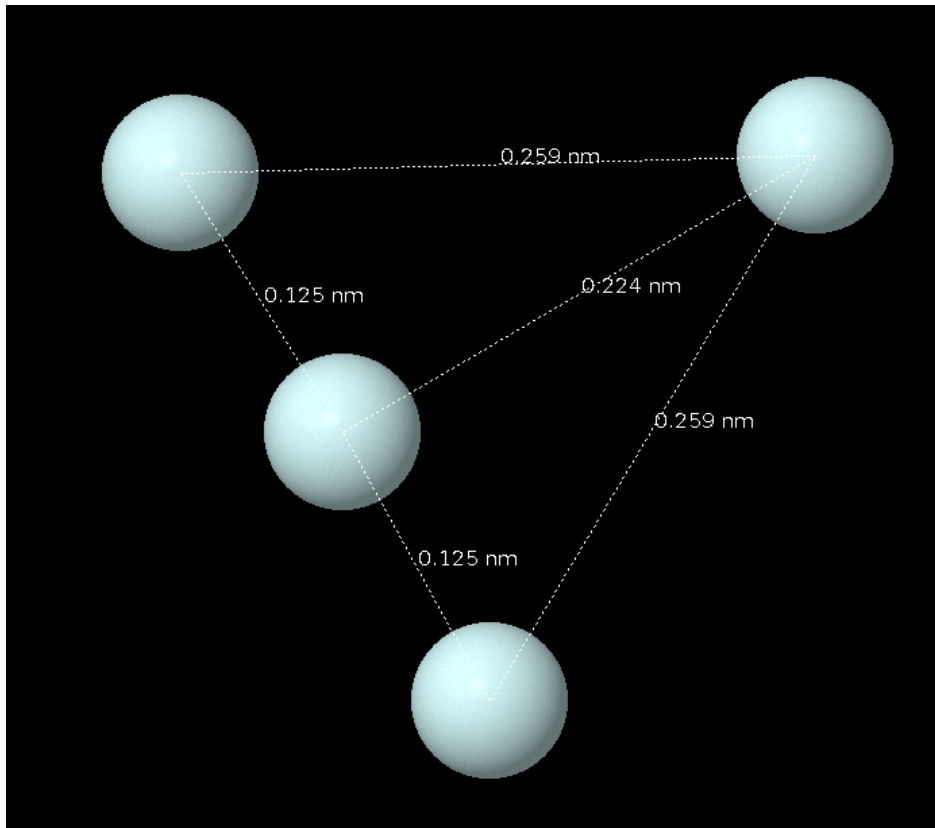
He₃⁺

Pro výpočet byla použita kvantověchemická metoda coupled clusters, augmentovaná báze quadruple-zeta. Byl nalezen jediný izomer.



He₄⁺

Pro výpočet byla použita kvantověchemická metoda coupled clusters, augmentovaná báze quadruple-zeta. Byly nalezeny dva izomery.



Závěr

Shrnutí

Byl vytvořen spolehlivý systém propojení simulačních a interakčních programů, který je univerzální a použitelný k optimalizaci pomocí programů libovolného dodavatele. Jsou také připraveny simulační programy schopné lokální optimalizace funkce a hledání více minim funkce.

Vytvořené programy byly použity k hledání strukturních izomerů iontových klastrů helia a argonu a bylo dosaženo dobrých výsledků. Program je tak připraven k hledání strukturních izomerů libovolných molekul s malým počtem atomů.

Výhledy do budoucna

Nejbližším plánem do budoucna je pokračovat v hledání strukturních izomerů He_n^+ a také dokončit simulační program založený na evolučních strategiích. Blízko je také spolupráce s prof. Ing. Ivanem Zelinkou, který vytvořil simulační program založený na metodě diferenciální evoluce.

Dlouhodobějším cílem je zaměřit se na hledání sedlových bodů. Dnes stále neexistuje zcela spolehlivá metoda, která by byla schopna sedlové body najít. Sedlové body jsou jednoznačně matematicky determinovány, problém je tedy pouze v simulační metodě. Zde je prostor pro univerzálnost propojení, protože můžeme vyzkoušet mnoho simulačních metod při minimální potřebě změn, bude stačit napsat nový simulační program a jeho dva interprety.

Použitá literatura a software

Vývoj programů:

- Midnight Commander

Zpracování výsledků:

- Jmol – zobrazování konfigurací molekul (XYZ souborů)
- Gimp – úprava finálních obrázků

Provedení výpočtů:

- Molpro quantum chemistry package, 2008

Použitá literatura:

- Evolučné algoritmy – Vladimír Kvasnička, Jiří Pospíchal, Peter Tiňo
- Úvod do počítačových simulací – Ivo Nezbeda, Jiří Kolafa, Miroslav Kotrla
- Fortran 90 Course Notes – AC Marshall, JS Morgan, JL Schonfelder
- Manuál Molpro - <http://www.molpro.net/info/current/doc/manual.pdf>

Seznam příloh

- Zdrojové kódy interpretů pro interakční program Molpro
- Simulační program v3
- Řídicí program
- Příklad start.ini