

Dr. Buttyán Levente bírálói kérdéseire adott válaszok

MTA doktora értekezés szerzője, címe:

Kertész Attila

Komplex, elosztott rendszerek viselkedésének elemzése

Először is szeretném megköszönni Dr. Buttyán Leventének, hogy időt és energiát szánt MTA doktori értekezésem alapos átolvasására, tanulmányozására, és megfogalmazta opponensi véleményét. Köszönöm az értekezéssel kapcsolatos hasznos észrevételeit és kérdéseit, amelyeket az alábbiakban válaszolok meg:

"A disszertáció 4 téziscsoportot tartalmaz. Az első téziscsoportban federált felhő rendszerek kialakítására és működtetésére használható új módszerek kaptak helyet. Ezek a Jelölt önálló tudományos munkásságának legkorábbi szakaszában született eredmények. A további téziscsoportokba IoT-felhő, IoT-köd-felhő, illetve blokklánc technológiát alkalmazó köd-felhő rendszerek modellezésével és elemzésével kapcsolatos új eredmények kerültek. Ezek a Jelölt önálló munkásságának későbbi szakaszaiban születtek, amikor már doktorandusz hallgatók témavezetésével és nemzetközi projektek munkacsomagjainak irányításával is foglalkozott, ezért természetes - és ezt a Jelölt is korrektil megjegyzi - hogy ezek az eredmények a doktoranduszaival és más kutatótársaival együtt végzett kollaboráció során születtek. Ez teljes mértékben megfelel a mai kutatási gyakorlatnak."

Válasz: Köszönöm az Opponens megerősítő véleményét.

"Fontosnak tartom kiemelni, hogy publikációk többsége mértékadó, a szakmai közösség által elfogadott minőségű és presztízsű folyóiratban illetve konferenciakiadványban jelent meg. Megjegyzem továbbá, hogy a publikációk egy töredék részéhez nem tudtam hozzáférni, tekintve, hogy azok nem nyílt elérésűek, és az őket tartalmazó digitális könyvtárakhoz sem nekem, sem munkahelyemnek (Budapesti Műszaki és Gazdaságtudományi Egyetem) nincs jelenleg előfizetése vagy hozzáférése. Ez azonban nem korlátozott abban, hogy a Jelölt eredményeinek lényegét megértsem, és azok jelentőségét megítéljem."

Válasz: Köszönöm az Opponens elismerő véleményét. Azt sajnálattal olvastam, hogy egyes publikációimhoz a kiadói adatbázisokon keresztül nem tudott hozzáférni az Opponens. A pályázati anyagban leadott önéletrajzomban a publikációk témakör alatt feltüntettem a tudománymetriai adatokat is tartalmazó kutatói közösségi profiljaim hivatkozásait, többek között a ResearchGate weboldali linket is (amely a honlapomról is elérhető). Ezen az oldalon keresztül megtalálható a nyílt hozzáféréssel nem rendelkező publikációim megjelenés előtti verziója; ezzel igyekszem elkerülni a hasonló eseteket.

"1. téziscsoport: Felhő föderációk vizsgálata

1. kérdés: Nekem úgy tűnik, hogy az 1.1 tézisben javasolt bróker architektúra megtervezése, valamint a kapcsolódó erőforrás monitorozás, jobbra mérnöki feladatok. Milyen tudományos jellegű kihívással szembesült ebben a munkában?"

Válasz: Elismerem, hogy az FCM architektúra megtervezése, és a SALMon monitorozó megoldással történő integrációja elsősorban mérnöki kihívásokkal rendelkező feladat volt. Emellett viszont hangsúlyozom, hogy a felhő rendszereket magasabb szinten vezérlő, meta-brókerező módszer, és a virtuális gépekbe telepített, különféle metrikákkal operáló szolgáltatófüggetlen felhő monitorozó módszer a megjelenés időpontjában új megoldásnak számított. A tudományos kihívás, amelyre választ ad a kidolgozott integrált megoldás, a dinamikus terheltségi információkkal alapról nem rendelkező (illetve azokat nem publikáló), több-felhős infrastruktúrákon történő hatékony alkalmazás futtatás.

"Az 1.2 tézisben a Jelölt felhő federációkban és IoT-kód-felhő rendszerekben jelentkező adatvédelmi problémák csoportosítására ad módszert és azt alkalmazza különböző használati esetekre, majd általános következtetéseket próbál megállapítani a fent említett rendszerek adatvédelmi problémáival kapcsolatban. Nekem a munka eredményeként született következtetések elég kézenfekvőnek tűnnek, akár az elvégzett elemzés nélkül is megállapíthatók. Pl. az egyik következtetés szerint, "amint a rendszerek komplexitása növekszik, azaz egyre több technológiát alkalmaznak, úgy gyengül a felhasználók ráhatása az adatkezelési folyamatokra, melynek következtében nagyobb felelősség hárul a szolgáltatókra"."

Válasz: Első ránézésre valóban kézenfekvőnek tűnik az idézett következtetés. Második olvasatra viszont egy hipotézis is felfedezhető benne, amit igazolni kell. Az 1.2-es tézis egy hosszú, 10 éves iterált, multidiszciplináris kutatássorozat eredményét jelenti. Először, 2011-ben az S-Cube EU FP7-es kiválósági hálózat kutatói csereprogramjában vettem részt a Tilburgi Egyetemen, amikor elkezdtük vizsgálni a több-felhős megoldásokban futtatható alkalmazások adatvédelmi kérdéseit, még az Európai Unió 1995-ös adatvédelmi irányelve alapján (eredménye: [180]-as publikáció az értekezésben). Az SZTE Állam- és Jogtudományi Karának bevonásával ezt követően is folytattuk a vizsgálatokat, és az IoT-Kód-Felhő rendszerekre történő kiterjesztéssel, már az uniós általános adatvédelmi rendelet (GDPR) alapján számos beazonosított adatkezelési eset jogi megfelelésének kérdését vizsgáltuk (eredménye: [181]-es publikáció az értekezésben). A bizonyítást a beazonosított esetek, és az azokat leíró esettanulmányok és elemzésük jelenti, amelyek alátámasztják az idézett állítást. Az eredményeket és tapasztalatokat beépítettük az SZTE TTIK és ÁJTK kapcsolódó kurzusaiba.

"2. kérdés: Volt-e az 1.2 tézisben végzett elemzésnek olyan eredménye, mely a Jelölt számára meglepetést okozott, vagy olyan új következtetéshez vezetett, ami a szisztematikus elemzés nélkül nem lett volna könnyen megállapítható?"

Válasz: Igen, az egyes beazonosított használati esetekben potenciálisan bekövetkező dinamikus szerepkör változásra derült fény a kutatás eredményeként, amely valóban meglepetést okozott. Az előző válaszban említett [181]-es cikkben, a 1995-ös adatvédelmi irányelvben bevezetett, és a GDPR-ban megerősített, valamint kibővített szerepkörök megjelenését vizsgáltuk meg hat konkrét használati esetben, amelyek az adott alkalmazások adatmozgásának lehetőségeit csoportosítják IoT-Kód-Felhő infrastruktúrák igénybevitelével. Az említett szerepkörök: adatkezelő, közös adatkezelők és adatfeldolgozó. Az esetlegesen bekövetkező adatvédelmi incidensekért az adatkezelő(k) felelős(ek). A megnevezett hat esetből háromban előfordulhat, hogy a kezdetben adatfeldolgozó szerepben részt vevő infrastruktúra szolgáltató később adatkezelővé válik, és így eltérő szabályok vonatkoznak rá, amire fel kell készülni, és megfelelően dokumentálni kell.

"3. kérdés: A fenti konkrét következtetés esetében, hogyan mérhető egy rendszer komplexitása, illetve a felhasználók ráhatása az adatkezelési folyamatokra, és milyen függvénnyel írható le a komplexitás függése az alkalmazott technológiák számától?"

Válasz: Véleményem szerint nem határozható meg általánosan egy ilyen összefüggést leíró függvény. Az előző válaszban említett hat eset meghatározásakor törekedtünk arra, hogy bővítsük az alkalmazandó technológiai megoldások körét az esetek rögzítésében (IoT → IoT+felhő → IoT+köd+felhő), ami bár növelte az alkalmazott rendszer komplexitását, mégis a bennük végrehajtható alkalmazások vagy igénybe vett szolgáltatások adatkezelési módjaitól függ a jogi megfelelés kérdése. Ennek következtében a rendszerben alkalmazott technológiák számának növekedésével nem feltétlenül nő a jogi megfelelés bonyolultsága. A felhasználók ráhatása szintén nagyon alkalmazásfüggő, és elsősorban az adatkezelés céljától és jogalapjától, valamint az alkalmazásfejlesztő döntéseitől függ. Így a ráhatást illetően a felhasználói biztonságtudatosság növelése a célunk, amit az is elősegít, ha jobban megismeri az átlag felhasználó az ilyen rendszerek viselkedését, és a bennük lezajló folyamatokat.

"Az 1.3 tézisben a Jelölt új optimalizációs algoritmusokat javasol felhő federációk adatközpontjainak energiahatékony működtetésére és azokat szimulációs módszerekkel elemzi. A javasolt módszer fő újdonsága, hogy a feladatok VM-ekhez történő allokációja során az energiafelhasználás és a futási idő együttesen kerül optimalizálásra, illetve a módszer lehetőséget ad az energiafelhasználás és a futási idő közötti trade-off-ok keresésére. Megjegyzem, hogy a disszertációban ezt a közös optimalizációt és trade-off keresést nem igazán szemlélteti a Jelölt."

Válasz: Valóban, területi korlátok miatt nem tudtam részletesen bemutatni az értekezés 1.3-as alfejezetében összefoglalt eredményeket, amelyeket részletesen az értekezés [101]-es hivatkozásában feltüntetett cikkben olvashatunk. Ebben a munkában vezettük be azt a Pliant rendszert alkalmazó döntési stratégiát, amely a felhő adatközpontokban elérhető virtuális gépeket kiszolgáló processzorok terheltségét és energiafogyasztási modelljét veszi figyelembe az alkalmazás komponensek gépekre történő allokálásához. Az energiafelhasználás és a futási idő közötti trade-off beállítása a Pliant módszerben alkalmazott paraméterek finomhangolásával történhet. Az energiafelhasználás csökkentésére a megfelelő paraméterek módosításával azt érhetjük el, hogy a feladat allokációk során nagyobb valószínűséggel választunk kisebb fogyasztású processzorral rendelkező virtuális gépet.

"4. kérdés: Összehasonlította-e a Jelölt a javasolt optimalizációs módszert más optimalizációs módszerekkel? Mit tart a javasolt optimalizációs módszer hátrányának?"

Válasz: Az értekezés 1.3-as alfejezetében tárgyalt Pliant rendszert alkalmazó döntési stratégiáról részletesen az értekezés [101]-es hivatkozásában feltüntetett cikkben olvashatunk. Ebben a munkában a Pliant módszer finomhangolt verzióit mohó algoritmusokkal hasonlítottuk össze. Ezek egy része az elérhető legkisebb energiafogyasztást és leggyorsabb futási időt hivatott mohó módon elérni (a baseline értékek meghatározásához), valamint készítettünk egy véletlenszerű VM választást megvalósító algoritmust, és egy olyan optimalizációs algoritmust is, amely 80%-os VM terheltség esetén újabb VM-et indít (a várakozási idő minimalizálására). Ezekkel az algoritmusokkal összevetve szemléltettük a Pliant módszer hatékonyságát.

A Pliant optimalizációs módszer alkalmazásának hátránya, hogy a paraméterek beállítása szakértelmet és időt igényel (azaz előzetes kísérletek futtatását és kiértékelését igényli). Egy jelenleg is folyó kutatás próbál megoldást találni erre a problémára, amelyben MI-alapú algoritmusokkal tervezzük automatizálni a Pliant módszerben alkalmazandó paraméterek beállítását.

"2. téziscsoport: IoT-felhő rendszerek vizsgálata

5. kérdés: Mit tekint a Jelölt a 2.2 tézis fő eredményének: az új, részleges szimulációs módszert vagy a MobIoTSim szimulátort? Ha a módszert, akkor miben látja annak fő tudományos értékét? Azaz milyen kihívások motiválták az új módszer kifejlesztését, melyeket nem lehetett megoldani más ismert szimulációs módszerekkel, illetve milyen elméleti problémákat kellett megoldani a módszer kifejlesztése során?"

Válasz: Véleményem szerint nem lehet egyértelműen szétválasztani a módszert és a szimulátort a tudományos megítélés szempontjából. A módszer, mint megközelítés, az azt, illetve annak működését demonstráló megvalósítás nélkül nem teljesértékű. A tudományos értéket a megközelítés újszerűsége és a motivációs igény adja. Nagyméretű IoT rendszerek tervezése, és főként az azokat felhasználó alkalmazások kifejlesztése nem lenne megvalósítható szimulációs megoldások nélkül. Az értekezés 2.1-es alfejezetében mutattam be a módszer kidolgozásakor a szakirodalomban fellelhető hasonló célú megoldásokat. Ezek egy része általános célú szimulátor volt, amelyek nem képesek valós alkalmazásokkal együttműködni, mások pedig túl speciális célú üzleti megoldások voltak, amelyek a tudományos közösség számára zárt hozzáférésűek voltak. A fő eredmény tehát a két megközelítést ötvöző módszeren alapuló megoldás. Az így létrejött MoBioTSim szimulátor előnye, hogy standard interfészekon keresztül tetszőleges IoT használati esetet megvalósíthat, amelyhez segédalkalmazásokat is kidolgoztunk (SUMMON és az átjáró szolgáltatás). Emellett a szimulátor nyílt forráskódú, és mélyebb szakértelem nélkül is könnyen felkonfigurálható.

A mérnöki kihívásokat az üzleti szolgáltatók (pl. IBM és Azure) és a standard kommunikációs megoldások (pl. MQTT és REST) interfészeinek támogatása jelentette. Az ezeken túlmutató elméleti problémát a megvalósítás alapjául szolgáló Android alkalmazásban a nagyszámú szimulált IoT eszköz hatékony kezelésének igénye jelentette, amelyet megfelelő szálkezeléssel értünk el ([102]-es a kapcsolódó publikáció az értekezésben).

"6. kérdés: Mik a főbb, tudományos jellegű kihívások, problémák egy privát átjáró szolgáltatás kifejlesztésében, melyekhez a state-of-the-art mérnöki módszereken túlmutató új módszerek szükségesek?"

Válasz: A MoBioTSim szimulátort kifejlesztésekor az IBM, majd az Azure felhő szolgáltatók kiegészítő IoT szolgáltatásaihoz kapcsoltuk, azaz szolgáltató-specifikus átjárókkal tudtuk alkalmazni. Ez kezdetben megfelelőnek tűnt, mivel az IBM megoldása nyílt forráskóddal rendelkezett. Ezt követően viszont azt tapasztaltuk, hogy az egyes szolgáltatók interfészei gyorsan elkezdtek változni, ami egyrészt a MobIoTSim-ben is kompatibilitási problémákat okozott, másrészt az üzleti átjárókat (és egyéb üzleti szolgáltatásokat) alkalmazó felhasználók körében is nehezen követhetővé tette azok folyamatos frissítését. Ennek a problémának a megoldása jelentette a fő kihívást az általános, privát (azaz üzleti szolgáltató-független) átjáró kifejlesztésében.

Szintén tudományos jellegű kihívással bír az átjáró által kezelendő IoT eszközök skálázhatóságának támogatása. Ezt egyrészt az eszközcsoportok bevezetésével oldottuk meg, amely gyorsabb és

átláthatóbb vizualizációt valósít meg, másrészt a szolgáltatás megvalósításakor a megfelelő architektúrális tervezéssel kívántuk támogatni a főbb komponensek Docker-alapú mikroszolgáltatásokba szervezésével (mely a továbbfejlesztett, specializált IoT gateway esetében lett kihasználva). További javulást érhetünk el az IoT eszközöktől érkező üzenetek kezeléséért felelős bróker megfelelő kiválasztásával. Ehhez kapcsolódóan az értekezés 2.2-es alfejezetében bemutatott teljesítménymérések adhatnak támpontot. Ezen túl egy jelenleg is folyó kutatásban próbálunk javítani az üzenetkezelés skálázhatóságán elosztott brókerek alkalmazásával.

"3. téziscsoport: IoT-köd-felhő rendszerek vizsgálata

7. kérdés: Minek köszönheti a DISSECT-CF-Fog szimulátor a jobb skálázhatóságát? Milyen árat kell ezért a skálázhatóságért fizetni? Mik a DISSECT-CF-Fog szimulátor skálázhatóságának határai?"

Válasz: A DISSECT-CF-Fog szimulátor jobb skálázhatóságát a hasonló célú, legelterjedtebben használt iFogSim szimulációs eszközzel összehasonlítva mutattuk meg. A hatékonyságbeli eltérést az említett eszközök alapjául szolgáló DISSECT-CF és CloudSim alapszimulátorok megvalósításában kell keresni. Egy virtuális gép allokációs algoritmus implementálásával és portolásával Mann az alábbi cikkben mutatott rá arra, hogy a CloudSim esetén a szimuláció során felhasznált adatok és információk tárolása több esetben nem elég hatékony, mivel gyakran az információ egymásba ágyazott kollektiókból (például Map-eket tartalmazó listák bejárásával) vagy más esetekben több interfész bevonásával nyerhetők csak ki:

- Z. Mann. Cloud simulators in the implementation and evaluation of virtual machine placement algorithms. *Software: Practice and Experience*, 48, 2017. DOI: 10.1002/spe.2579

Ez a független tanulmány azt is kimutatta, hogy a DISSECT-CF használata esetén a szükséges információk tárolásának közvetlenebb módja kevesebb memória felhasználásával jár. A DISSECT-CF fejlesztése során a kód optimalizálásra is nagy figyelmet fordított a szerző a JProfiler eszköz segítségével.

Ezt az optimálisabb memóriafelhasználást támasztja alá az értekezés 3.1-es alfejezetének utolsó két bekezdésében hivatkozott [119]-es számú publikációnk is, amelyben a DISSECT-CF-Fog és az iFogSim eszközöket egy közös IoT-Köd-Felhős használati eset szimulációjának segítségével hasonlítottuk össze, szintén a JProfiler eszközzel. Az eredmények azt mutatták, hogy a megegyező skálájú szimulációk esetén a DISSECT-CF-Fog kevesebb memóriát használ, és összességében a dinamikus memória (heap) mérete 200 MB alatt marad, míg az iFogSim esetében majdnem eléri a 400 MB-ot. A létrehozott objektumok számát tekintve közel négyszer annyi objektum kerül példányosításra az iFogSim-mel végzett kiértékelés során. Mindazonáltal fontos hangsúlyozni, hogy a szimulátorok belső felépítése és működése nem minden esetben feleltethető meg teljes mértékben egymásnak. A DISSECT-CF-Fog előnyét a fő vetélytársával szemben az is jól szemlélteti, hogy a hasonló kísérletek futtatásával történő összehasonlítás során az iFogSim hasonló tesztelési környezetben az ezres nagyságrendű eszközszerű eseteket már nem volt képes lefuttatni, míg a DISSECT-CF-Fog-gal több százszázalékos kísérleteket is el tudtunk végezni.

A szimulátor skálázhatóságának a határait a DISSECT-CF-Fog szimulátort futtató számítógép erőforrás specifikációja adja. Az általunk futtatott szimuláció esetén, amelyben egy 24 órás időjárás előrejelző alkalmazás viselkedését modelleztük, 3 felhő/kód csomópont, illetve 100 000 IoT eszköz és 500 000 szenzor működését definiáltuk. Ez a szimuláció megközelítőleg 1 órás futási időt eredményezett egy Intel Core i5-7300HQ 2.5GHz, 8GB RAM, 64-bites Windows 10 operációs

rendszerrel működő számítógépen. Ez volt a legnagyobb skálájú kísérlet, amit le tudtunk futtatni (a nagyobb skálájúakat egy óra után leállítottuk).

"4. téziscsoport: IoT-blokklánc-köd-felhő rendszerek vizsgálata

8. kérdés: Mennyivel jobb a 4.2 tézisben javasolt módszer és protokoll az irodalomban található más konzisztencia elemző módszerekhez és adattovábbító protokollokhoz képest? Történt-e összehasonlító elemzés, s ha igen, mik voltak annak főbb konklúziói?"

Válasz: Az értekezés 4.2.3-as alfejezetében mutattam be a konzisztenciaelemzési módszerünket, amelyről részletesebben a [96]-os publikációban olvashatunk. Ebben elméleti összehasonlítással vizsgáltuk meg a szakirodalomban fellelhető megközelítéseket. Az elosztott főkönyv konzisztenciáját a rendszerben egy adott pillanatban jelen lévő forkok (azaz az egyes csomópontokon eltérő tartalommal rendelkező láncok) száma határozza meg. Az ilyen forkok megjelenésére hatással van egy blokklánc rendszer számos tulajdonsága, például a benne lévő csomópontok, és azok szomszédainak száma. A szakirodalomban meglévő megoldások csak arra a kérdésre adtak választ, hogy mekkora a valószínűsége annak, hogy egy adott forgatókönyvben megjelenik egy fork (láncváltózat). A mi javasolt módszerünk ezen túlmenően a válasz számszerűsítését célozta meg azáltal, hogy arra a kérdésre ad választ, hogy hány fork jelenhet meg a láncok lehetséges maximális számához viszonyítva, és ez hogyan változhat egy adott blokklánc paramétereinek változtatásával. Tehát a mi módszerünk újdonsága az volt, hogy a FoBSim szimulátor segítségével a csomópontok szomszédainak számát az egyéb beállítási paraméterektől függően úgy optimalizálhatjuk, hogy az maximális konzisztencia értéket eredményezzen. Megjegyzem, hogy a munkánk során empirikusan megmutattuk, hogy az összes vizsgált paraméter közötti kapcsolat nem lineáris.

A konzisztenciaelemzési módszerünk alkalmazásának, azaz a FoBSim-mel történő kiértékelésének eredményeit követően javasoltuk a DONS dinamikus szomszédválasztó optimalizációs protokollt a fent említett kritikus tényezőnek, azaz a csomópontonkénti szomszédok számának további optimalizálására. Vagyis a javasolt protokoll a konzisztenciaszint csökkentése nélkül képes meghatározni a hálózat logikai felosztásával azt a gráfot (minimális feszítőfát), amit alkalmazhatunk az adatok továbbításához. Az értekezés 4.2.3-as fejezetében összegeztem a [31]-es publikációban részletezett mérési eredményeinket, amellyel megmutattuk, hogy a DONS módszer alkalmazása képes csökkenteni az adatgenerálástól az adat teljes hálózaton történő szétterítéséig eltelt időt. A szakirodalmi összehasonlítást illetően szintén a [31]-es publikációban mutattuk meg, hogy a DONS-ban alkalmazott AnoLE alprotokoll a szakirodalomban elérhető DMST-hez képest nagyobb szintű adatvédelmet biztosít. Emellett a DONS protokollt összehasonlítottuk a szakirodalomban elterjedten használt RNS és RTT-NS módszerekkel. A DONS alkalmazásának eredményei jelentős javulást mutattak a blokkvégeességi (block finality) idő és a kommunikációs komplexitás tekintetében.

Szeged, 2024. május 10.


.....
Kertész Attila