

BME Építészmérnöki Kar

Tudományos Diákköri Konferencia
2023

A geometria szerepe kiselemes rendszerek optimális kiosztásában

Készítette:

Domonkos Nikolett

Martinek Júlia Zita

Konzulens:

Dr. Fehér Eszter

BME Morfológia és Geometriai Modellezés Tanszék

Tartalomjegyzék

1. Bevezetés.....	2
2. Problémafelvetés	2
2.1. Jelenlegi megoldások	3
2.2. BIM és parametrikus tervezés.....	3
2.3. (Vágott) csempék előregyártásának lehetőségei	4
2.4. Korábbi tanulmányok eredményei	4
2.5. Tanulmányunk célja	5
3. Módszer.....	6
3.1. Tervezési algoritmus.....	7
3.4. Optimalizálás.....	10
4. Esettanulmány.....	11
4.1. A vizsgált példa	11
4.2. Optimalizálási esetek	13
4.3. Megoldások és diszkusszió	15
5. Összefoglalás	18
6. Köszönetnyilvánítás.....	19
7. Hivatkozások	20

A geometria szerepe kiselemes rendszerek optimális kiosztásában

1. Bevezetés

Az építőipar a globális gazdaság egyik meghatározó ágazata, amely kulcsfontosságú szerepet játszik bolygónk gazdaságában. 2022-ben a globális GDP több mint 14%-át tette ki, és előrejelzések alapján további növekedés várható [1]. Az építkezések kezdeti fázisa általában a költségbecslést foglalja magában, ahol a várható költségek és bevételek kiszámítása zajlik, természetesen az előbbi csökkentésére és az utóbbi növelésére törekedve. Az építési projektek során az anyagok beszerzése jellemzően az összberuházás legalább 40%-át teszi ki [2], ezért az anyagfelhasználás csökkentése kiemelt szempont a költségek csökkentése érdekében. Napjainkban az élők munkája jelentősen növekedett, így a kivitelezők különböző módszerekkel igyekeznek a szükséges élők munkáját csökkenteni. Épp ezért előtérbe került az üzemi előregyártás, ám a padlóburkolatok vágása terén ez még kevésbé elterjedt, leginkább helyszínen történik ez. Ezenkívül egyre fontosabbá válik az építőipar fenntarthatóságának javítása is, amely magában foglalja a termelt hulladék mennyiségének csökkentését a pontos anyagmennyiség számításával és csak a szükséges mennyiség beszerzésével.

2. Problémafelvetés

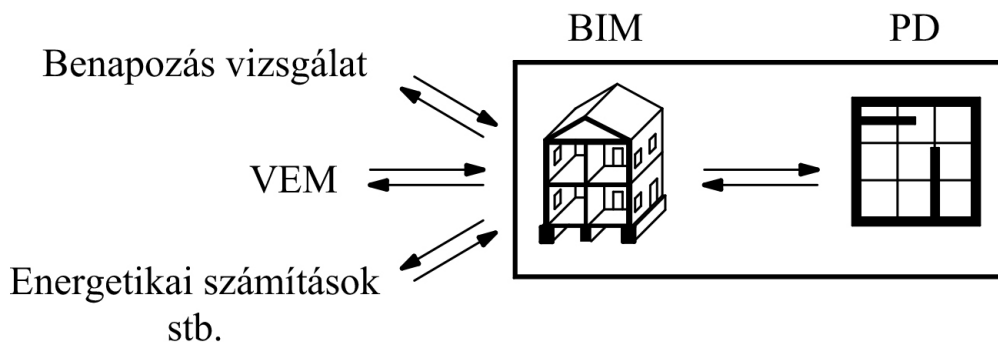
Az építkezések egyik fő hulladékforrása a kiselemes rendszerek alkotóelemei, kiselemes padlóburkolatok, gipszkarton lapok, falazóblokkok stb. Csak az elemes padlóburkolatok adják jellemzően a termelt hulladék 5-15%-át [3]. Korábban ezen kiselemes rendszerek kiosztását általában az építészek vagy az építési helyszínen dolgozó szakemberek készítették el, rendszerint kézi módszerekkel, ökölszabályok alapján, a tervezési folyamat befejezése után, de a mai fejlődő világban már léteznek erre a problémára jobb megoldási lehetőségek.

2.1. Jelenlegi megoldások

A digitális forradalom térnyerésével az építőiparban digitális megoldások születtek, amelyek segítenek csökkenteni az anyagfelhasználást. Ding és Xiao [4] például rámutattak, hogy az intelligens tervezés akár 40%-kal is csökkentheti az anyagfelhasználást az építkezéseken. Ennek ellenére a kiselelemes rendszerek 2D-s optimalizálása még hiányolja az erre megfelelő eszközöket. A hagyományos tervezőprogramok, mint például a Revit [5], Archicad [6] vagy SketchUp [7], általában nem rendelkeznek az optimalizációhoz szükséges számítási kapacitással. Az ilyen modulok hozzáadása jelentős hardverigényt róna a tervezési szoftverekre. Léteznek a problémára specializálódott kiegészítő szoftverek, mint a Cutting Planner Pro [8], a Tile Hatch Builder for Autodesk CAD [9] vagy a Roombook Extension for Revit [10], ám ezek nem veszik figyelembe a vágott és hulladékként eldobott csempék felhasználásának lehetőségét.

2.2 BIM és parametrikus tervezés

A Building Information Modeling (BIM) modell lehetővé teszi a különböző szakágak összehangolását és az építészeti tervekhez kapcsolódó adatok kezelését [11]. Ám a BIM elsősorban adatgyűjtésre alkalmas, és számításokhoz kiegészítő programra van szüksége: esetünkben a parametrikus tervezés eszközeire, mely képes különböző bemeneti adatok alapján optimalizálni a várt megoldást. A parametrikus tervezés során a tervező elsősorban egy algoritmust készít, amely egy folyamatosan frissülő modellt hoz létre, és ennek segítségével finomítja tovább az eredeti tervét (1. ábra).



1. ábra: A BIM és a parametrikus tervezés

2.3. (Vágott) csempék előregyártásának lehetőségei

Napjainkban a padlóburkolatok elkészítésekor jellemzően egy közelítő számítást követően helyben hozza meg a döntést az elhelyezésről a szakember. A csempék vágása is helyben történik, eseti jelleggel, vagy legfeljebb egy-két kis helyiség számára előre elkészítve, kézi szerszámokkal. A mesteremberek a lehető legkevesebb anyag felhasználására a tapasztalatuk, illetve ökölszabályok alapján törekednek.

Azonban az utóbbi évtizedekben jelentős előrelépések történtek az építőipari előregyártás [21] és a lézer-, illetve plazmavágás területén [22, 23]. Így lehetővé vált a kerámia-csempék gépesített, precíz és pontos vágása. Ez üzemi környezetben elvégezhető, és így az építési helyszínre megfelelően konszignált, pontosan kiszámolt mennyiségű csempe érkezik. Nem csak az alapanyag beszerzésének anyagi vonzata csökkenthető ilyen módon, de lényegesen egyszerűbbé válhat a deponálás, csökken a helyszíni kivitelezéshez szükséges idő, és nem szükséges a keletkező hulladék tárolásáról és elszállításáról gondoskodni.

2.4. Korábbi tanulmányok eredményei

A kiselemes padlóburkolatok optimális kiosztásának problémájával először egy 2021-es tanulmány foglalkozott [12]. Ebben Wu és társai egy olyan parametrikus tervezési megoldást dolgoztak ki, mely a padlóburkolat elemek eltolásával keresi meg a minimálisan felhasználandó darabszámot egy adott lakás burkolása során. A tanulmány számításba veszi a vágott burkolatok felhasználását is, így a hulladék aránya jelentősen alacsonyabb, mint ami az általános, analóg tervezés során jellemző. Egy évvel később Wu és társai további tanulmányban [13] foglalkoztak a problémával. Ekkor felvetették a nem elmozdítható, fix berendezési tárgyak kérdéskörét is, valamint íves felületekre is optimalizáltak. Ezenfelül a problémát *pareto* optimalizálási feladatként fogták fel, így nem csak egy-egy aspektusban keresték a legjobb megoldást, hanem olyan eredményt kerestek, mely (ugyan bizonyos kompromisszummal) minden tekintetben (anyagfelhasználás, hulladékarány, költség stb.) a legmegfelelőbb. 2023-ban két tanulmány is folytatta a korábbi kutatásokat. Wu és társai [18] korábbi algoritmusukat továbbfejlesztve még szokatlanabb alakú alaprajzokra keresték a megoldást, Xu és társainak algoritmusai [19] pedig nem csak négyzetes kiosztásokkal próbálkozott, hanem a burkolásban elterjedt

egyéb mintázatokkal is (pl. kötésben fektetett padlólapok, halszálla mintázat stb.). Ebben a tanulmányban a legelső publikáció [12] kiinduló alaprajzát használták, így demonstrálva a korábbi eredményeken való javítást.

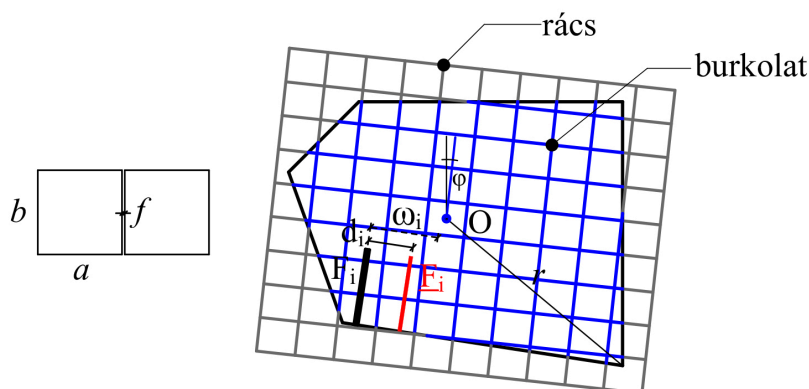
2.5. Tanulmányunk célja

Bár a már korábban kifejlesztett algoritmusok [12, 13, 18, 19] jelentős előrelépést jelentenek a padlóburkolatok (és egyéb síkra vetíthető elemek) kiosztásának problémáján, még mindig megőrzik az analóg módszerek egyik fő hibáját: az optimalizációs folyamatot a tervezés befejezése után készítik el, minden korábbi tervezési döntést véglegesnek tekintve. Az alkalmazott BIM-alapú tervezés lehetővé teszi, hogy már a tervezés kezdetén vizsgáljuk és optimalizáljuk ezt a problémát. Így a lehető legjobb eredmény elérése érdekében bizonyos változtatható tervezési paramétereket állapítottunk meg. A falak vagy egyéb határoló szerkezetek helyének tervezése a legtöbb építési projektben bizonyos keretek közt esetleges, kis mértékben való elmozdításuk nem befolyásolja az eredeti koncepciót, azonban jelentős hatásuk lehet a beruházás költségeire mind a hulladékmennyiség mind pedig az elemek megmunkálási igénye szempontjából. Így modellünkben a lehető legjobb csempekiosztás érdekében ezek pontos elhelyezkedését változóként kezeltük. További hiányosságként azonosítottuk az elemek forgathatóságának hiányát, melyet szintén implementáltunk a tanulmányban. A korábbi eredményekkel [12, 18] való összevethetőség érdekében a tanulmányban az először Wu és társai által használt alaprajzon teszteltük az algoritmusunkat, az általuk elért legjobb eredményeket kiindulási helyzetként kezelve.

Épületeink a mai világban először valamilyen modellező program segítségével vizualizálódnak, a legcélszerűbb, ha egy BIM programban történik mindez. Célunk volt, hogy egy 3D modellként meglévő épület alaprajzát be lehessen hivatkozni egy parametrikus tervező programba, mely optimalizálja a tervező által kijelölt területen az adott padlóburkolat kiosztását, majd ezen kiosztást vissza is lehessen küldeni a BIM modellbe. Kutatásunk esetében a BIM program az Archicad [6], a parametrikus szoftverünk pedig a Rhino-beli [17] Grasshopper volt [16].

3. Módszer

A matematikában léteznek olyan problémák, ahol egy síkot különböző sokszögekkel hézagmentesen fednek le, ezt nevezik a “sík csempézése” feladatnak [14]. Ha szabályos sokszögekkel történik mindez, akkor szabályos csempézésnek nevezzük, ha pedig többféle szabályos sokszöget használunk, de meghatározott szabály szerint, akkor félig szabályosnak. Tanulmányunkhoz kapcsolódik még a “2D bin packing problem” [15], magyarul ládapakolási probléma, ahol különböző formájú elemeket úgy rendeznek el adott “tárolóegységekben”, hogy a lehető legkevesebb ilyen egységet használjanak fel. Wei és társai [15] síkban vizsgálták a különböző méretű téglalapok adott méretű téglalap egységekbe való elrendezését és ennek minimalizálását. Az elemes padlóburkolatok kiosztását, bár a valóságban három-dimenziós elemek, síkbeli problémaként is felfoghatjuk, mivel a vastagsági méretük sokkal kisebb, mint a másik két kiterjedésük. Kutatásunk során egy adott sokszög körvonalú sík és egy sík kitöltés, vagyis a burkolati elemek sorolásából álló rács optimális elhelyezkedését vizsgáljuk egymáshoz képest úgy, hogy a rácsból levágott elemek és azok maradéka is újra felhasználható legyen. Így a lehető legkevesebb burkolóelemet használjuk fel a lehető legkevesebb hulladék keletkezése közben.



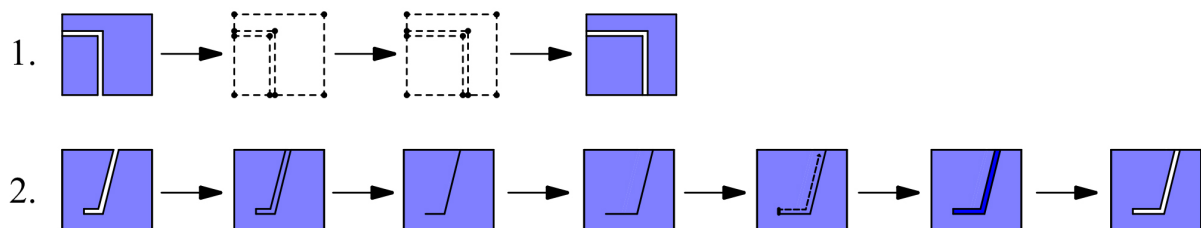
a	a burkolatelem szélességi mérete	r	a poligon középpontjától legtávolabb eső csúcs távolsága
b	a burkolatelem hosszúsági mérete	N_v	vágott elemek száma
f	fugaméret	N_c	vágtatlan, egész elemek száma
F_i	fal	N_k	azon egész elemek száma, melyre a vágott elemeket rendeztük
E_i	elmozgatott fal	$N=N_c+N_k$	összes felhasznált elem
ω_i	a fal mozgási tartománya	R	hulladék arány
d_i	a fal tényleges elmozdulása		
O	a körvonal poligon középpontja		
φ	a rács elforgatásának szöge		

2. ábra: A használt jelölések és elnevezések

3.1. Tervezési algoritmus

A 2. ábrán látható jelöléseket és kifejezéseket használtuk a módszer leírásánál. A Grasshopper [16] szkript az Archicad [6] modellből átveszi az épület falainak és a burkolandó területnek a geometriáját. A vastagabb tartófalak helyét fixnek vettük, viszont a felhasználó megadhatja, hogy az alaprajz belső részén található válaszfalak közül melyek pozícióját változtathatja a program, melyeket az optimalizáció során külön is kezel. Az előbb említett mozgatott válaszfalakat az általunk vizsgált példában az alapján választottuk ki, hogy az elmozgatásuk okoz-e síkegyeztetési problémákat vagy sem. Így végül azok helyét parametrizáltuk, amelyek elmozgatása nem vonja maga után tartófal, vagy több másik, hozzá nem kapcsolódó válaszfal mozgatását is. A válaszfalak mozgatása a burkolandó terület módosítását vonja magával. Az elmozgatott válaszfalak egy új sokszöget hoznak létre, mely összemetsződéseit újra ki kell szerkeszteni, hogy egy zárt görbét kapjunk.

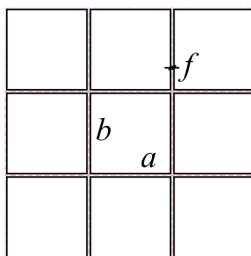
A folyamat során sok esetben maga a csempézendő terület nagysága is változik. A mozgatás típusától függően két módszert implementáltunk (3. ábra). Az első lehetőséget azoknál a válaszfalakkal használjuk, melyekhez kapcsolódnak más falak. Az utóbbiak elmozdítása nem cél, viszont az elmozdított válaszfalhoz való nyújtás és rövidülés igen. A csempézendő terület körvonalának pontjai lekérésével kiválasztjuk azon pontokat, melyek meghatározzák a változtatandó válaszfalakat. Merőlegesen csatlakozó falaknál elég, ha fal síkjára vett normálvektorral mozgatunk, ahol viszont a falak ferdén kapcsolódnak, ott a ferde fallal párhuzamos vektorral mozgatjuk el az ahhoz tartozó pontot. Az új csempézendő terület körvonalát megkapjuk, ha az eredeti pontokból eltávolítjuk a korábbi falak helyét meghatározó pontokat és a maradékot hozzáadjuk az elmozdított pontokhoz, ezáltal egy új poligont hozunk létre. A másik lehetőséget azoknál a válaszfalakkal alkalmaztuk, melyek nem alkotják a burkolandó terület



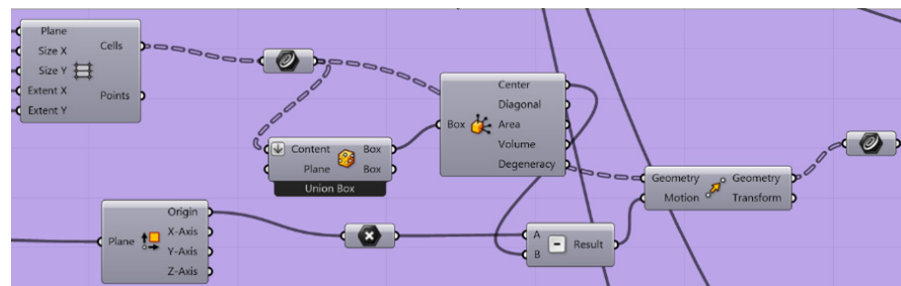
3. ábra: Variációk falak eltolására

kontúrját, hanem csak benyúlnak a felületre. A csempézendő terület felületét “begyógyítjuk” a válaszfal eredeti helyén lévő felületével úgy, hogy a két felületet egyesítjük. Az válaszfalnak vesszük az azt meghatározó vonalait, és a változtatni kívánt falrészt elmozgatjuk egy, a síkjára merőleges normálvektorral, majd a hozzá kapcsolódó fal vonalát meghosszabbítjuk. A fal új felületét megkapjuk, ha a meghosszabbított és az elmozgatott falvonalat összemetszünk, egyesítjük, majd a vastagságával megegyező vonallal végigseprünk rajta. Végül az elmozdított válaszfal felületét kivágjuk a “begyógyított” felületből. Az algoritmus vége egy körvonal, mely megadja az optimalizálandó területet. Ezután elkészítjük a csempézendő területen kiosztani kívánt burkolati rácsot. Az adott burkolatnak ismerjük a szélességi (a) és hosszúsági méreteit (b), illetve meghatározhatjuk a tervezett fugaszélességet (f). A rácsot ezen adatok segítségével hozzuk létre.

A rács képzésénél a csempék méretébe a fugaméretet (f) is bele kell számolni, ugyanis a burkolat közötti hézagokat a rács létrehozása után hozzuk létre. Ezáltal a burkolat a és b méreteihez hozzáadjuk a fugaszélességet (f), tehát egy rácsbéli elem oldalméretei $a + f$ és $b + f$ lesznek. Ezen méretekkel hozzuk létre a rácsot téglalapokat hézagmentesen helyezve egymás mellé (4. ábra).

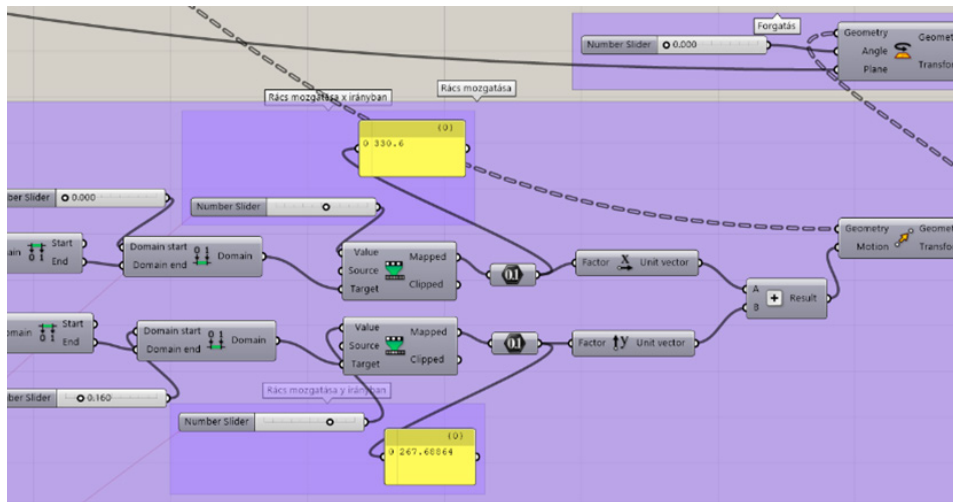


4. ábra: A rács geometriája



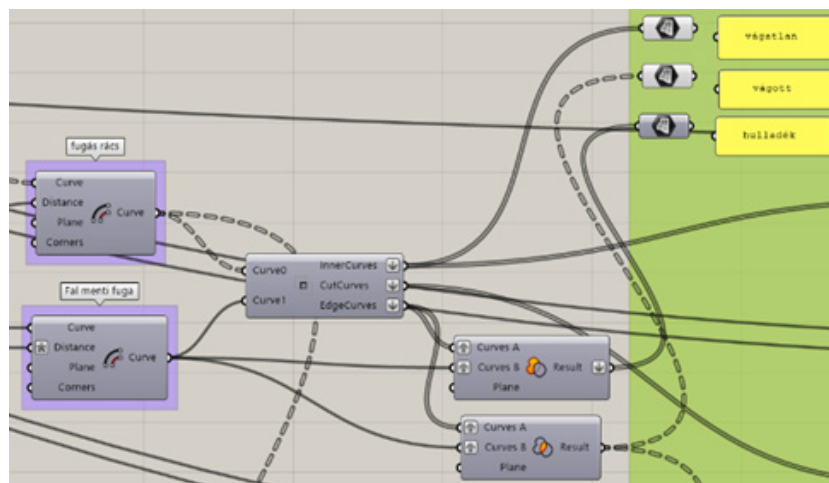
5. ábra: A rács létrehozása és eltolása

A rácsot áthelyezzük a középpontjánál fogva a csempézendő terület középpontjába (5. ábra). Ezután függőleges és vízszintes irányban elmozgathatóvá, illetve középpontja körül elforgathatóvá tesszük. A szükséges számítási kapacitás csökkentése érdekében kedvező, ha a rács nem feleslegesen nagyobb, mint a lefedendő poligon. Ehhez meghatározzuk a lefedendő poligon körvonalának azon pontját, mely a rács középpontjától a legmesszebb található (r). A rács méretét úgy állítjuk be, hogy a középponttól minden irányban legalább $r + \max(a, b)$ szélességű legyen. A rács mozgathatóságának mértékét a csempék mérete adja, vízszintes és függőleges irányban is maximum egy csempényi mozgás megengedett, így az összes lehetőséget lefedve (6. ábra). A forgatást a csempék eltolása után végezzük.



6. ábra: A rács mozgathatósága

A rácsban lévő hézagokat az elemek központi eltolásával (*Offset*) hozzuk létre, az eltolás mértéke a fugaszélesség fele, $f/2$, mivel a folyamat során minden rács elemet eltolunk. Ilyenkor csak a rácson belüli hézagokat hozzuk létre, de a valóságban a falak mellé is készül fuga, ezért a csempézendő terület körvonalát is el kell tolni (*Offset*) f -fel a belső irány felé. Az eltolat körvonalat és az új rács rácsot összemetszünk (*Intersection*), majd az így kialakult burkolati elemeket szétválogatjuk. Ha a körvonalon belül helyezkedik el az elem és teljes, akkor vágatlan, ha körvonalon belül, de nem teljes, akkor vágott, ha a körvonalon kívül esik és nem teljes, akkor a valóságban nem kerülne elhelyezésre, ezért a számolásból eltávolítjuk (7.ábra).



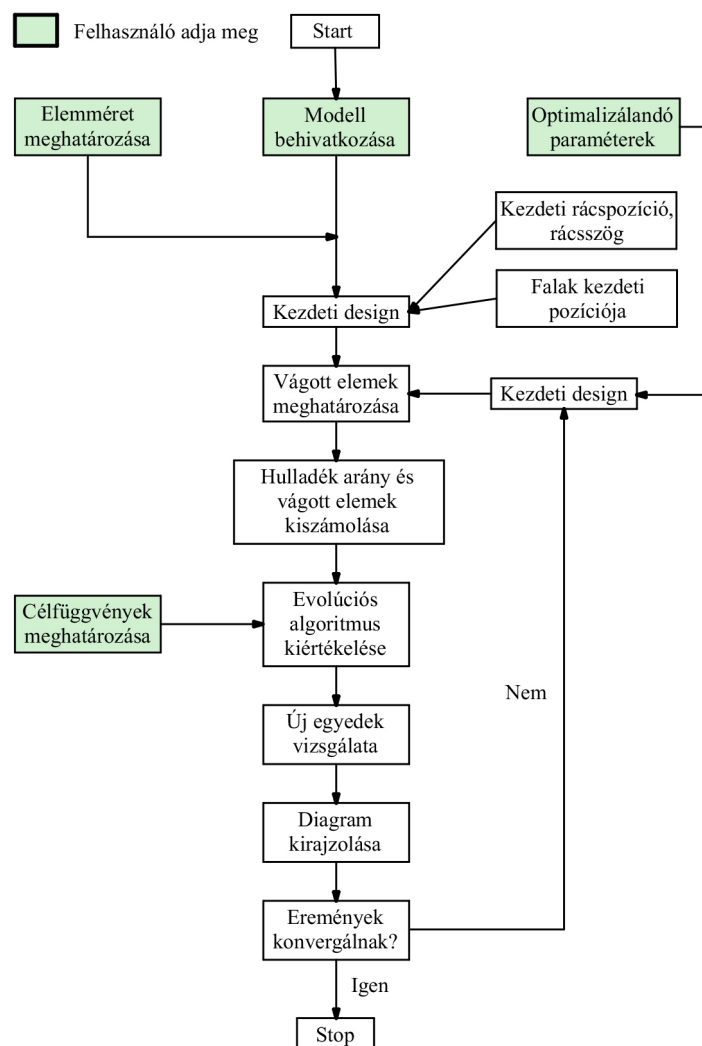
7. ábra: A különböző elemek szétválogatása

A következő jelentős lépés a vágott elemekből adódó hulladék csökkentése. Az építési gyakorlatban gyakori az így létrejövő hulladék eldobása, esetleg egy egész csempéből kettő vagy három elem kivágása, ám ezek spontán döntések sorozata következtében történnek, és nem előre eltervezett folyamatok. Tanulmányunkban a vágott elemeket egész elemekre helyeztük,

melynek optimális kiosztását az OpenNest plugin [20] biztosítja. Ez a plugin gyakorlatilag a már korábban említett ‐ládapakolási‐ problémát oldja meg. Szükséges megadni ‐ládákat‐, azaz megfelelő számú egész burkolatelemet, illetve a vágott elemeket, melyeket ezekre szeretnénk elhelyezni úgy, hogy minél kevesebb egész elemre legyen szükség. Az OpenNest [20] használatával egy elemből akár több vágott elem is előállítható, melyek precíz elhelyezése és vágása lehetőséget ad a legjobb anyagkihasználásra. A pluginban továbbá állítható a ráhelyezett elemek egymástól és a csempe szélétől való távolsága, így bármilyen későbbi vágási metódushoz finomhangolható. Így a vágott elemekhez szükséges egész elemek száma jelentősen csökken.

3.4. Optimalizálás

Az optimalizálást a Galapagos pluginnal [24] végeztük, mely az ún. evolúciós algoritmus segítségével optimalizálja azokat a paramétereket, amelyeket megadunk. Bemenetébe a változtatható paramétereket kötöttük be, a kimenetét, pedig az adott célfüggvény értékére irányítottuk. Az evolúciós algoritmus az egyes paraméterek kombinációjából variációkat, azaz egyedeket hoz létre, melyeket kiértékel a célfüggvény segítségével, majd ez alapján új egyedeket generál, figyelembe véve a korábbi iteráció alatt legkedvezőbb eredményre vezető paraméterkombinációkat (8. ábra).



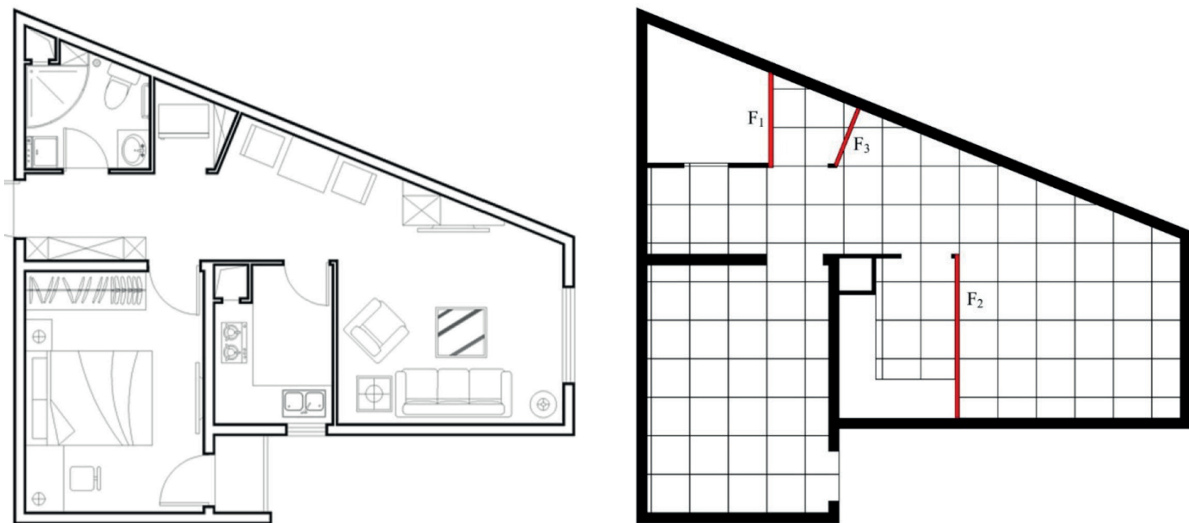
8. ábra: Az optimalizáció lépései

Az optimalizálási folyamatot futtató számítógép specifikációi: AMD Ryzen 7 6800H with Radeon Graphics 3.20 GHz processzor, 16 GB RAM és Windows 11. A futtatási idő körülbelül 2-2,5 óra/optimalizáció volt, amely nagyjából 40-50 ciklust jelentett.

4. Esettanulmány

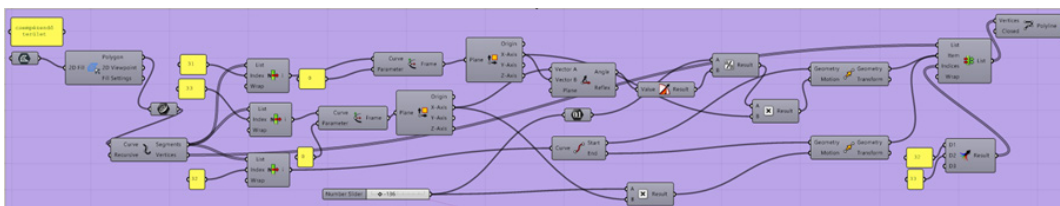
4.1. A vizsgált példa

A tanulmány elején is tárgyalt cikkekben [12,19] található alaprajzot (9. ábra) vettük alapul. Ezzel a konkrét példával Wu és társai a 2021-es [12], illetve Xu és társai a 2023-as cikkükben [13] foglalkoztak. A 2021-es tanulmányban [12] vettek egy 60×60 cm²-es padlóburkolatból álló rácsot, ahol az elemek között lévő fugát 6 mm-esnek vették. Ezen az alaprajzon ezt a rácsot x és y irányban mozgatták, és ennek függvényében minimalizálták az összes, illetve a vágott elemek számát. A 2023-as cikkben [13] Xu és társai ugyanezzel az alaprajzzal vizsgáldták, csak ők a mozgáson kívül különböző mintázatú burkolatkiosztásokkal optimalizáltak. Tanulmányunknál elsősorban az előbb említett kutatások közül a 2021-es [12] cikket vettük alapul, az abban található alaprajzon vizsgáldtunk, illetve eredményeinek javítását tűztük ki célul. Az eddigiekhez képest az általunk használt módszer során valamivel lazítunk az előbb említett cikk paraméterein. Nem csak a rácsot mozgatjuk, hanem az alaprajz geometriáját is minimális mértékben változtatjuk, illetve magát a rácsot is forgatjuk.

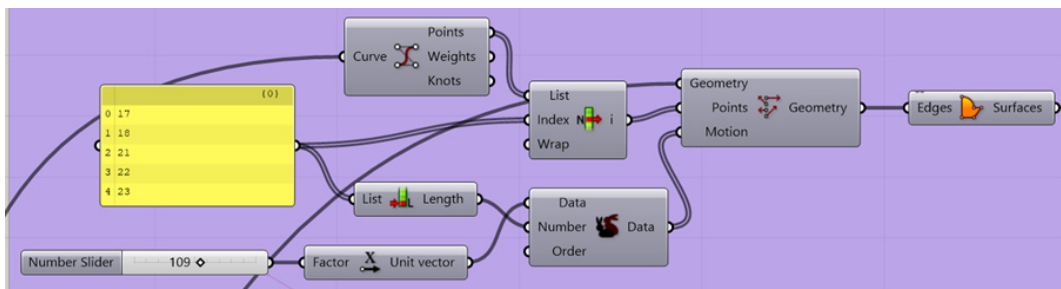


9. ábra: Az eredeti alaprajz, valamint a kijelölt elmozdítandó falak

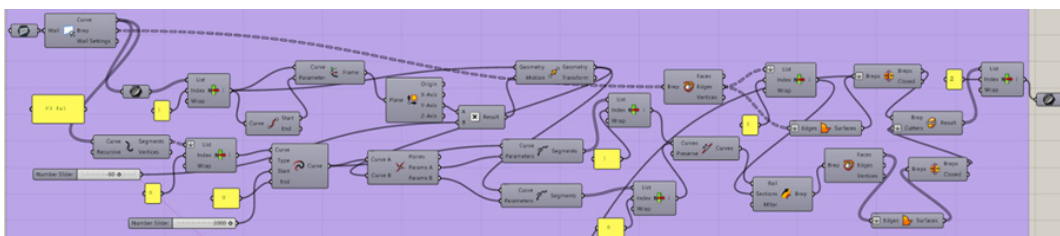
Az 9. ábrán látható a terület, melyet a burkolandó területnek vettünk, valamint a válaszfalakat (F_1 , F_2 , F_3), melyeket mozgatunk az optimalizálás során. Az F_1 és F_2 falak mozgatásánál a Módszer fejezetben tárgyalt első módszert használtuk, míg az F_3 -nál a másodikat. A mozgatás mértékét úgy határoztuk meg, hogy az építészeti koncepciót ne változtassa, így az F_1 fal mozgatási tartománya milliméterben $w_1 = \{-200; 200\}$, az F_2 falé $w_2 = \{0; 200\}$, az F_3 falé $w_3 = \{-200; 0\}$, a tényleges elmozdulás pedig rendre d_1 , d_2 , d_3 . A mozgatás tartományát és irányát az alaprajzi elhelyezkedés alapján határoztuk meg (10., 11., 12. ábra).



10. ábra: A csempézendő terület körvonalának megadása, F1 fal mozgatása



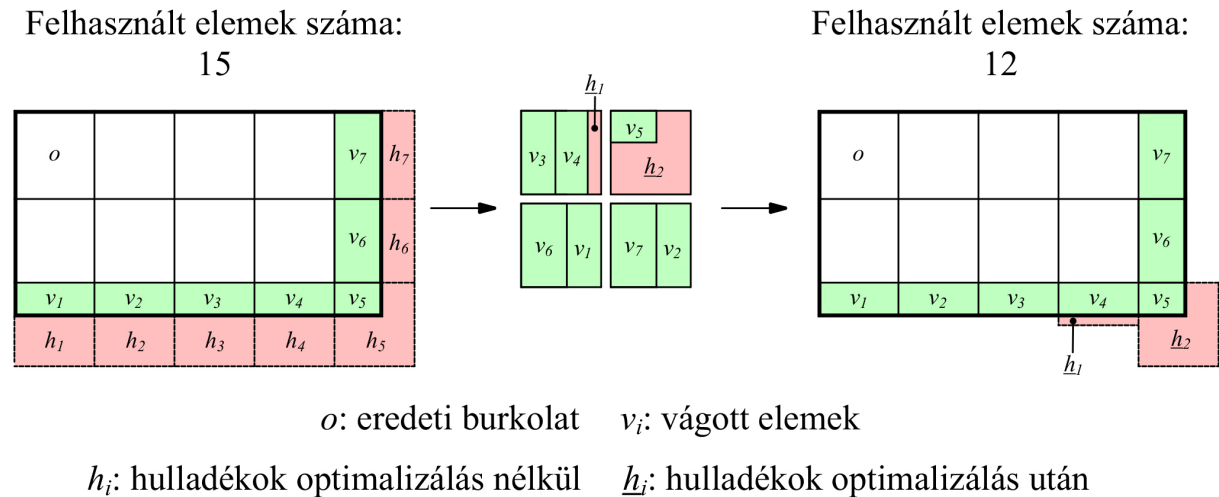
11. ábra: F2 fal mozgatása



12. ábra: F3 fal mozgatása

Wu és társai cikkéből [12] csak a burkolandó terület nagysága derült ki, pontos méreteket nem adtak meg, így az onnan vett ábra, illetve optimalis burkolat kiosztások alapján modelleztük meg Archicadben [6] a tárgyalt alaprajzot. A Grasshopper [16] szkript onnan vette át a burkolandó terület, mely körvonalának pontjai mozgatásával az F_1 és F_2 válaszfal változtatható, illetve az F_3 falat, melynek felületét aztán kivonjuk az addig generált burkolandó felületből. Összegezve, amit a Grasshopper [16] megkap az Archicadtól [6] az a csempézendő terület körvonala, illetve az F_3 fal. Ezekkel a bemeneti adatokkal készíti el az algoritmus az optimalizálás során változtatott alaprajzot és készít egy görbét, melyen belül mozgatja és egyes esetekben forgatja a rácsot. Az általunk használt rács $60 \times 60 \text{ cm}^2$ -es rács 6 mm -es fugával, amely bár nagyobb, mint amelyet a

valóságban készítene, azonban a 2021-es [12] cikkhez igazodtunk, ahol ezzel a kialakítással vizsgáloádtak. Az így létrejött vágott elemeket az OpenNest plugin [20] újra kiosztja úgy, hogy a lehető legkevesebb egész elemre férjen rá (13. ábra).



13. ábra: A hulladék lehelyezése a lehető legkevesebb egész elemre

4.2. Optimalizálási esetek

Kezdetben kijelöltünk négy pontot, melyeket, amennyiben a padlóburkolatot a hagyományos módszerrel készítik, a burkoló szakember kiindulópontként kezelhetne. Ezeket X eseteknek neveztük. Számoláskor ezeknél az eseteknél egy lehetséges intervallumot határoztunk meg a hulladékmennyiség kapcsán. Szakembertől függ, hány vágott elemet tud egy csempéből kivágni. Így meghatároztuk a lehető legkedvezőbb esetet, az intervallum másik végét pedig az képzi, ha minden vágott elemet külön csempéből állít elő. Már csak a hulladék felhasználásával jelentős megtakarítást lehet elérni, a burkolat elhelyezésének optimalizálása nélkül.

Az optimalizálás során két esetet különböztetünk meg Wu és társai vizsgálatai [12] alapján. A cikkben először az összes felhasznált elemet optimalizálták, tehát az abból számolt hulladékarányt is minimalizálták, ez volt az A eset, a B esetben pedig a vágott elemek számát minimalizálták. A hulladékarányt a cikkben [12] publikált $R_k = (N \cdot A_b) / A_t - 1$ képlettel számoltuk, ahol R_k a hulladék arány, N az összes felhasznált burkolatelem, A_b egy burkolatelem területe, A_t pedig az egész burkolandó alaprajzi terület. Ez csupán egy közelítő képlet, mivel nem veszi figyelembe a fugákkal kitöltött területet. A valódi arányt a következő képlettel lehet kiszámolni: $R_v = (N \cdot A_b - A_v) / A_t$, ahol R_v a valós hulladékarány, A_v a vágott elemek száma. Tekintettel

arra, hogy a korábbi tanulmányokban [12, 13, 18, 19] a hulladékarányt az első módszerrel számították, a továbbiakban mi is R_k értékkel dolgozunk.

Az A eset kedvező anyagfelhasználás tekintetében. B esetben a vágott csempék számát igyekeztünk csökkenteni, mely jelentősen csökkentheti a kivitelezés során az élőmunka-igényt, a szükséges időt, illetve a vágási folyamatot is, így közvetetten az anyagiakat is. Ezeket a célfüggvényeket lehet kombinálni is, ha például árat állítunk be magának a burkolatnak, az elemek kiviteli helyszínen való lerakásának vagy a vágási munkának. A 2021-es cikkben [12] ezekkel is vizsgáldtak, azonban tanulmányunkban, mivel az említett kutatásban [12] már eleve egy optimalizált kiosztást vizsgáltak, az A esetnél a hulladék arány minimalizálását tűztük ki célul, illetve a vágott elemek számának csökkentését.

Mindkét esetben négy opciót vizsgáltunk. Az első esetben a korábbi tanulmányokhoz [12, 13, 18, 19] hasonlóan csupán a rács pozícióját mozgatjuk (A_1, B_1). Ezután a második esetben a rács pozícióján kívül a rács forgatásának szögét is változtatjuk (A_2, B_2). A harmadik esetben már a lefedendő poligon körvonalát is változtatjuk, azaz eltoljuk a kiválasztott falakat (A_3, B_3). Végül a negyedik esetben mindhárom fejlesztést alkalmazzuk: a rács mozgatásával, elforgatásával és a falak eltolásával kerestük a legjobb megoldást (A_4, B_4). Mivel nem volt elérhető számunkra a pontos alaprajz, így a burkolandó terület és burkolatkiosztás alapján közelítettük az alaprajzot. Ugyanazokkal a beállításokkal közel azonos vágott és vágatlan elemszámot sikerült elérnünk (1. táblázat), mint Wu és társai [12].

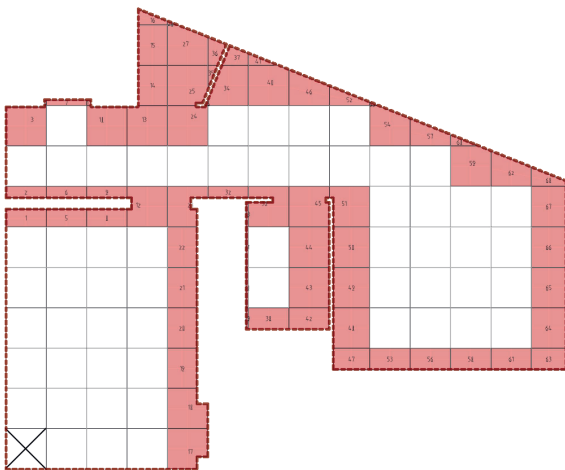
	vágatlan elem	vágott elem	összes elem	hulladékarány	terület változása	hulladék területe (m ²)	d1 (mm)	d2 (mm)	d3 (mm)	φ (rad)	teljes terület (m ²)	valós százalék arány
X1 elrendezés	58	68	101-126	2.45%-27,81%	-	0,87-9,87	-	-	-	-	35,49	4,71%-27,81%
X2 elrendezés	55	67	104-122	5.49%-23,75%	-	1,95-8,43	-	-	-	-	35,49	7,75%-23,75%
X3 elrendezés	60	63	104-123	5.49%-24,77%	-	1,95-8,79	-	-	-	-	35,49	7,81%-24,77%
X4 elrendezés	59	68	104-127	5.49%-28,83%	-	1,95-10,23	-	-	-	-	35,49	8.76%-28,83%
A1 elrendezés	62	79	99	0,42%	-	1,00	-	-	-	-	35,49	2,81%
A2 elrendezés	61	79	99	0,42%	-	0,99	-	-	-	0,001	35,49	2,79%
A3 elrendezés	60	79	98	0,17%	-0,76%	0,90	-136	109	-60	-	35,22	2,56%
A4 elrendezés	63	82	99	0,12%	0,31%	0,90	74	0	-45	4,709	35,60	2,53%
B1 elrendezés	60	63	103	4,48%	-	1,95	-	-	-	-	35,49	8,76%
B2 elrendezés	60	63	103	4,48%	-	1,95	-	-	-	0	35,49	8,76%
B3 elrendezés	57	61	106	7,54%	-0,30%	2,68	-3	1	-3	-	35,48	9,78%
B4 elrendezés	57	61	106	7,54%	-0,30%	2,68	-3	1	-3	0	35,48	9,78%

1. táblázat: Megoldások a különböző esetekben

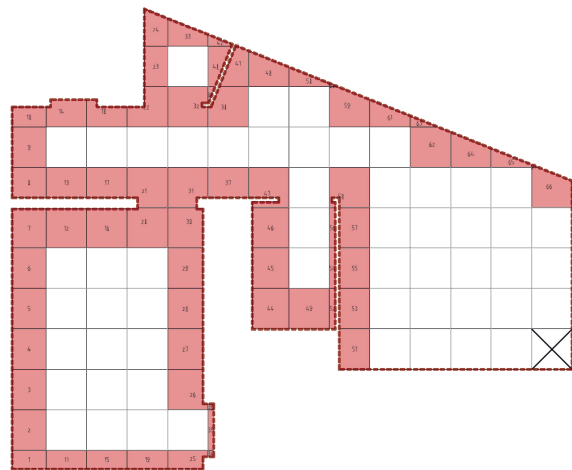
4.3. Megoldások és diszkusszió

A vizsgált opciók értékelésénél (1. táblázat) figyelembe vettük a vizsgált alaprajz területének változását és a falak helyének legnagyobb változását (amennyiben a falak helyzete változtatható volt), valamint a hulladék összterületét is. Így látszik, hogy milyen sokrétű a probléma, amivel szemben állunk, hiszen minden kivitelezésen más prioritások kerülhetnek előtérbe, más kiosztás megfelelő egy idővel takarékoskodó építkezésen, valamint egy pénzzel takarékoskodón, esetleg egyéb szempontrendszer követő helyszínen.

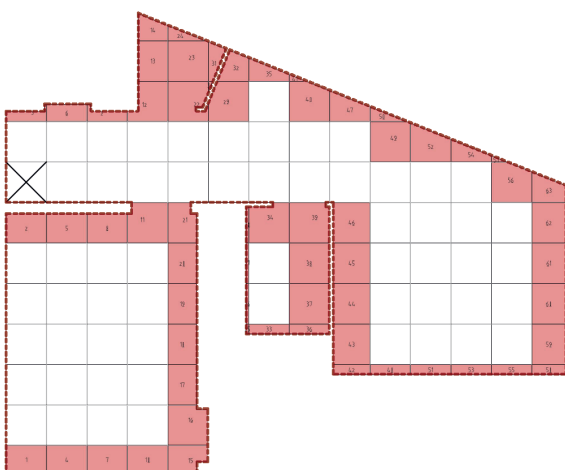
X esetekben, amikor a hagyományos kiosztás logikáját követtük, meghatároztunk egy-egy kezdő teljes elemet (X-szel jelölve), majd ezektől indulva elhelyeztük a padlóelemeket (14-17. ábra).



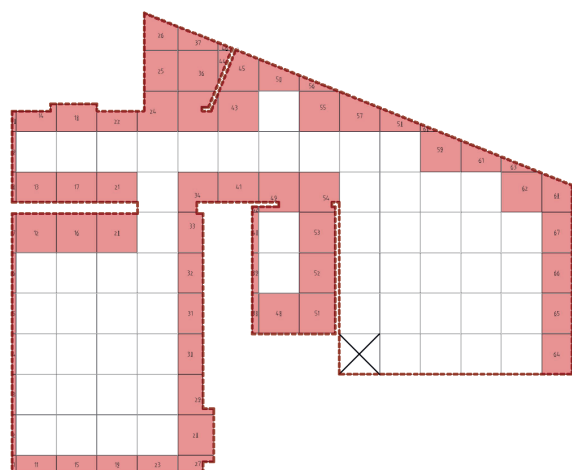
14. ábra: X1 eset csempekiosztása



15. ábra: X2 eset csempekiosztása



16. ábra: X3 eset csempekiosztása

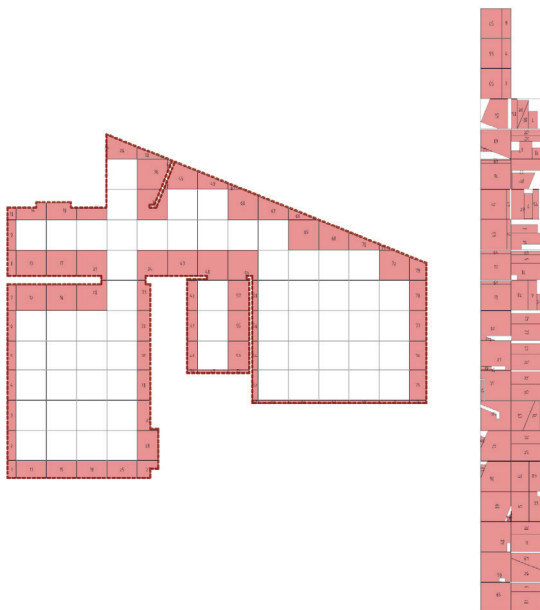


17. ábra: X4 eset csempekiosztása

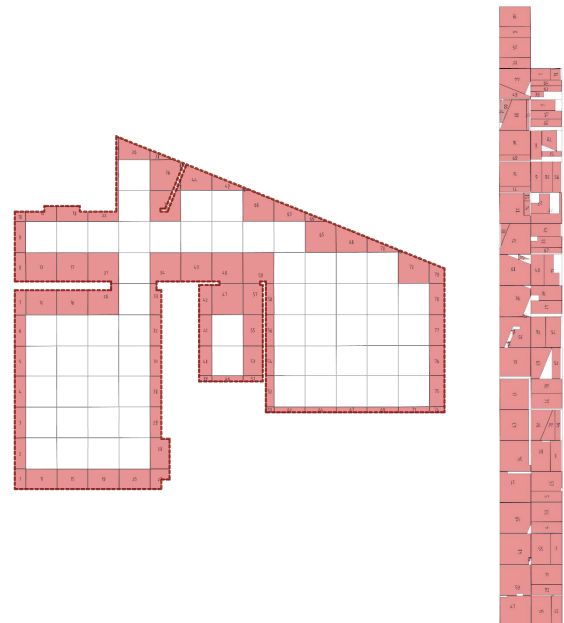
Az A esetekben a hulladékarány csökkentését tűztük ki célul. Először elvégeztük az A1-es optimalizálást (18. ábra), mikor csak a rácsot mozgattuk. Feltételezhetően az alaprajz

geometriájának apró eltérései miatt sikerült a korábbi cikkben [12] található eredménynél egy jóval kedvezőbb megoldást találnunk, mely az összes elem számát 2-vel csökkentette, a hulladékarányt pedig közel 2%-kal. A rács fogatásának (A2, (19. ábra)) hozzáadásakor ezek az eredmények marginálisan javultak. Mivel a rács szögelfordulása igazán kicsi, ezért ez az eredmény akár az A1 esetnek is betudható. Az evolúciós optimalizáció hátránya, hogy nem vizsgálja meg az összes esetet, ezért nem mondható biztosra, hogy az adott optimalizációs folyamat során a legjobb megoldást adja vissza, ahogyan az a jelen esetben is bebizonyosodott. Az A3 (20. ábra) esetben vizsgáljuk először a falak eltolhatóságát, először a rács szögét fixnek tekintve. Már ekkor jelentős, 0,25%-os javulást figyelhetünk meg a hulladékarányban, mely további 0,06%-kal csökken az A4 esetben (21. ábra), mikor a rács foroghat is.

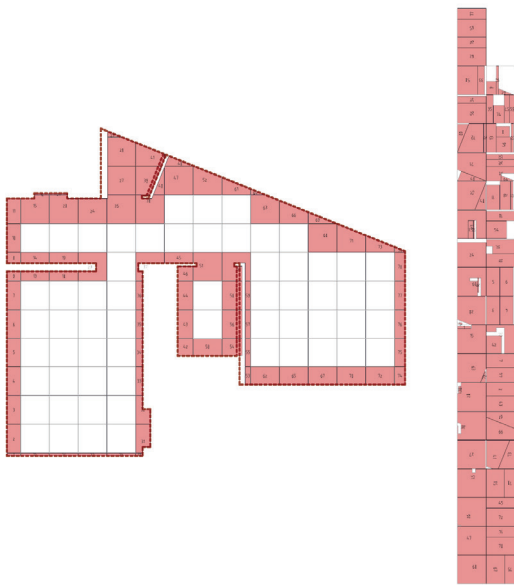
A falak elmozdításával a helyiség területe is változik. Az A3 esetben a falak legkedvezőbb helyzete és az eredeti helyük távolsága 6 és 14 cm közé esik, a helyiség területe 0,76%-kal csökken. Az A4 esetben, ugyan a hulladékarány, illetve a hulladék összterülete is csökken, a helyiség mérete mégis 0,3%-kal nő, a falak elmozdulása pedig maximum 7,5 cm. Ez azzal magyarázható, hogy összességében több elem kerül felhasználásra (98 helyett 99), és több vágott elem is keletkezik, ám ezeket kevesebb hulladék keletkezésével tudja egész elemekre rendezni. A valós hulladékarány minden esetben csökkent. Látható, hogy A4 esetben a legalacsonyabb a hulladékarány, ugyanakkor a vágott elemek száma, melyre a B esetben optimalizáltunk, a legmagasabb.



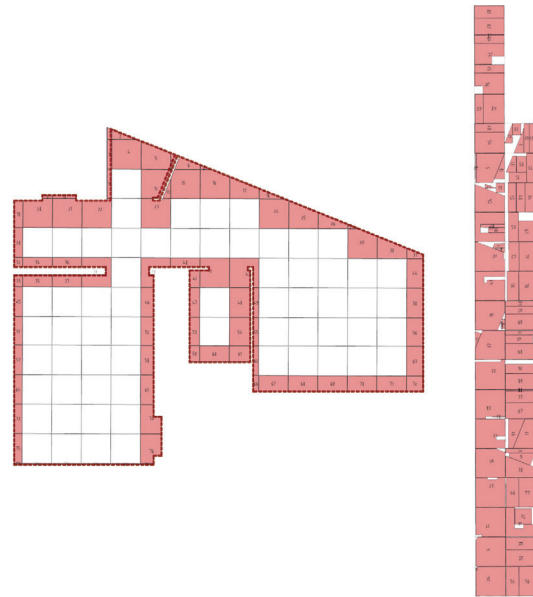
18. ábra: A1 eset csempekiosztása (bal) és a vágott elemek konszignációja (jobb)



19. ábra: A2 eset csempekiosztása (bal) és a vágott elemek konszignációja (jobb)

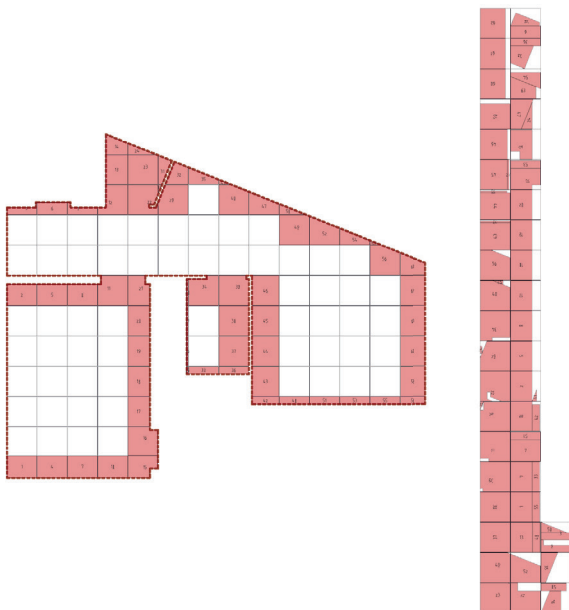


20. ábra: A3 eset csempekiosztása (bal)
és a vágott elemek konszignációja (jobb)

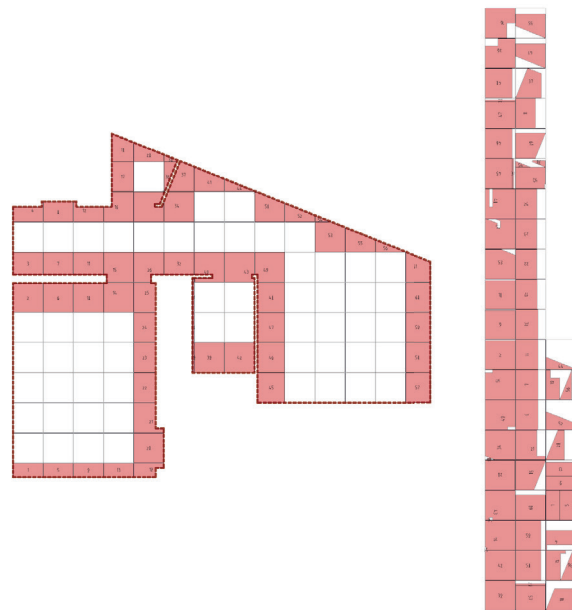


21. ábra: A4 eset csempekiosztása (bal)
és a vágott elemek konszignációja (jobb)

B esetekben a vágott elemek számának csökkentése volt a cél. B1 (22. ábra) állapotban, kizárólag a rács mozgatásával 63 vágott elemet sikerült elérni. Forgatással, B2 (22. ábra) esetben nem tapasztaltunk változást, ahogyan B4 esetben sem. Így megállapítható, hogy a kiosztás elfordítása nem jelent javulást a vágott elemek szempontjából. A falak mozgatásával B3 és B4 esetben (23. ábra) is javítható volt az eredmény. A vágott elemek számának csökkenésével a falak fél cm-t sem mozognak, a helyiség összterülete pedig csak elhanyagolható mértékben csökken. A hulladékarány és az összes elem száma nőtt a különböző esetekben. B4 esetben, ugyan ekkor a legkevesebb a vágott elem, a hulladékarány és az összes elem száma a legmagasabb a vizsgált esetek közül. Ebből is látszik, hogy a kiválasztott két eset egyidejű teljesítéséhez egy *pareto* optimalizációs feladatot kellene megoldani.



22. ábra: B1, B2 eset csempekiosztása (bal)
és a vágott elemek konszignációja (jobb)



23. ábra: B3, B4 eset csempekiosztása (bal)
és a vágott elemek konszignációja (jobb)

5. Összefoglalás

Összességében megállapíthatjuk, hogy jelentős anyag- és költségmegtakarítás lehetséges a padlóburkolatok lefektetésének területén a parametrikus tervezés bevonásával a BIM munkafolyamat korai szakaszába. Amennyiben az épület tervezését komplex egészként fogjuk fel, és az általában a tervezési szakasz végén megkezdődő munkafolyamatokat (pl.: padlóburkolati elemek konszignációja) már korábban vizsgáljuk, lehetséges a hulladék arányát a jellemző 5-15%-ról akár 2,5%-ra is csökkenteni, melynek ára kizárólag egy-két válaszfal maximum 20 centiméteres eltolása. A vágott elemek számát is jelentősen lehetséges csökkenteni, mely időt és munkát takaríthat meg egy kivitelezés során.

A tanulmányunk készítése közben több érdekes kérdés is felmerült bennünk, melyeket érdemes lehet tovább vizsgálni:

- A padlóburkolatok geometriája hihetetlenül sokszínű, így érdemes lehet egyéb elrendezéseket is vizsgálni.
- Nem csak a padlóburkolatok kiseleemes rendszerek egy építkezésben: álmennyezetek, homlokzatburkolati elemek, zsaluzatok és kőműves szerkezetek is lehetnek ilyenek. A módszer 3D-s problémákra is megoldást adhat, ha például falazóblokkokat szeretnénk kiosztani úgy, hogy minél kevesebb vágott elemet használjunk fel. Így ezeknél és még több szerkezetnél is érdemes lenne az optimalizáció elvégzése, és erre különböző módszerek kidolgozása. Ugyan egyedül a padlóburkolati elemek nem feltétlenül jelentős részei a költségvetésnek, amennyiben minden kiseleemes rendszer optimalizálása megtörténik, az jelentős hatást gyakorolna egy-egy építkezés összköltségvetésére.
- Bizonyos nem látszó rendszereknél (pl. falazatok) van lehetőség a kiosztást többféle méretű elemmel pl. fél- háromnegyed elemekből létrehozni. Érdekes geometriai probléma, hogy a rendelkezésre álló elemekből hogyan lehet elvégezni a kiosztást úgy, hogy minél kevesebb vágott elem legyen.
- Amennyiben a beruházás költségeit szeretnénk optimalizálni, úgy szükség van a problémában szereplő mennyiségek súlyozására. Legegyszerűbb esetben az elemek megmunkálásának az árát és az elemek anyagárát lehetne figyelembe venni.

- A falak eltolásának maximumát közelítőleg határoztuk meg. Érdeemes lenne egy hatástanulmány elvégzése, mely vizsgálja, hogy egy-egy válaszfal eltolása milyen hatással lehet az egész épületre.

Nem kérdéses, hogy a jövőben a parametrikus tervezés és a BIM használatának elterjedése alapjaiban fogja megváltoztatni az építőipart. Fontos felismernünk az ezekben a technológiákban rejlő potenciált, és megtalálni azokat a szegmenseket, ahol a lehető legnagyobb változást hozhatják: legyen ez az elemes padlóburkolatok lefektetésének optimalizálása, vagy bármilyen más részfeladat.

6. Köszönetnyilvánítás

Szeretnénk megköszönni Dr. Fehér Eszternek a konzulensi munkáját és végtelen segítőkészségét, illetve Dr. Hegyi Dezsőnek egy épületlátogatáson való részvételt, melyen megfigyelhettük az építőanyag optimalizálást a gyakorlatban is.

7. Hivatkozások

1. Company, T. B. R. (2023, July 12). Global Construction Industry Forecast - Market Size, Growth Rate And Leading Region, By The Global Market Model. [Www.prnewswire.com](https://www.prnewswire.com/news-releases/global-construction-industry-forecast---market-size-growth-rate-and-leading-region-by-the-global-market-model-301875304.html). <<https://www.prnewswire.com/news-releases/global-construction-industry-forecast---market-size-growth-rate-and-leading-region-by-the-global-market-model-301875304.html>>
2. Oseghale, G. E., Adetooto, J. D., & Oseghale, B. (2021). Evaluation of Material Management Methods in Selected Building Construction Sites in Abuja and Lagos State, Nigeria. *American Journal of Mechanical and Materials Engineering*, 5(1), 11-17.
3. Katz, A., & Baum, H. (2011). A novel methodology to estimate the evolution of construction waste in construction sites. *Waste management*, 31(2), 353-358.
4. Ding, T., & Xiao, J. (2014). Estimation of building-related construction and demolition waste in Shanghai. *Waste management*, 34(11), 2327-2334.
5. Autodesk, Inc. (2000a). Autodesk Revit [Software]. <https://www.autodesk.com/products/revit/overview?term=1-YEAR&tab=subscription>
6. Graphisoft. (2023). Archicad 26 (5002 INT) [Software]. <https://graphisoft.com/resources-and-support/loads?section=update&localization=HUN&type=FULL&version=26&platform=windows>
7. Trimble Inc. & @Last Software, Google. (2000). SketchUp [Software]. <https://www.sketchup.com>
8. TubakuroSoft, Cutting Planner Pro, 2021 [Last access 2022-3-20]; Available from: <https://cutting-planner-pro.apponic.com/>.
9. Thien Thanh, Tile Hatch Builder, 2022 [Last access 2022-07-01]; Available from: https://apps.autodesk.com/ACD/en/Detail/Index?id=5325235265949514346&appLang=en&os=Win32_64.
10. Lab., Autodesk, Roombook Extension For Revit, 2022 [Last access 2022-03-20]; Available from: <https://knowledge.autodesk.com/support/revit>.

11. Ehsan Kamel, Ali M. Memari, Review of BIM's application in energy simulation: tools, issues, and solutions, *Autom. ConStruct.* 97 (2019) 164–180, <https://doi.org/10.1016/j.autcon.2018.11.008>.
12. Wu, S., Zhang, N., Luo, X., & Lu, W. Z. (2021). Intelligent optimal design of floor tiles: A goal-oriented approach based on BIM and parametric design platform. *Journal of Cleaner Production*, 299, 126754.
13. Wu, S., Zhang, N., Luo, X., & Lu, W. Z. (2022). Multi-objective optimization in floor tile planning: Coupling BIM and parametric design. *Automation in Construction*, 140, 104384.
14. Domok, D. (n.d.). Síkkitöltések szabályos sokszögekkel. DomokDavid.web.elte.hu. Retrieved October 30, 2023, from <https://domokdavid.web.elte.hu/csempezes/>
15. Wei, L., Oon, W. C., Zhu, W., & Lim, A. (2013). A goal-driven approach to the 2D bin packing and variable-sized bin packing problems. *European Journal of Operational Research*, 224(1), 110-121.
16. Rutten, D. R. & Robert McNeel&Associates (TLM, Inc.). (2021). Grasshopper 3D (1.0.0007.) [Software]. <https://www.grasshopper3d.com/>
17. Robert McNeel&Associates (TLM, Inc.). (2021a). Rhinoceros 3D (6 SR35 (6.35.21222.17001)) [Software]. <https://www.rhino3d.com/download/archive/rhino-for-windows/6/latest/>
18. Wu, S., Lu, W. Z., Qin, C., Wen, B., Wu, D., & Xiang, Y. (2023). A goal-oriented planning approach for two-dimensional cutting components in architectural design: Coupling BIM and Parametric design (PD). *Journal of Building Engineering*, 107156.
19. Xu, Y., Wang, J., & Yang, Z. (2023). Generic goal-oriented design for layout and cutting of floor tiles. *Automation in Construction*, 152, 104903.
20. Vestartas, P. V. (2021). OpenNest (1.5.1) [Software]. <https://www.food4rhino.com/en/app/opennest>
21. Murari, S. S., & Joshi, A. M. (2017). Precast construction methodology in construction industry. *Available at SSRN 3496019*.

22. Kanyilmaz, A. (2019). The problematic nature of steel hollow section joint fabrication, and a remedy using laser cutting technology: A review of research, applications, opportunities. *Engineering Structures*, 183, 1027-1048.
23. Kalvettukaran, P., Keshari, P., & Misra, D. (2023). Parametric study of laser cutting of glazed ceramic tiles. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 45(6), 333.
24. Rutten, D. R. (2019). Galapagos [Software]. In ADDON (1.0.0007.). <https://grasshopperdocs.com/addons/galapagos.html>