

روشی برای حل مساله جدول برنامه هفتگی مدارس با استفاده از رویکرد تجزیه در برنامه ریزی ریاضی

امین وحیدی منفرد^{۱*}، علیرضا علی احمدی^۲

۱- دانشجوی دکترا مهندسی صنایع دانشگاه علم و صنعت، تهران، ایران

۲- دانشیار دانشگاه علم و صنعت، تهران، ایران

رسید مقاله: ۲۸ فروردین ۱۳۹۲

پذیرش مقاله: ۲۶ شهریور ۱۳۹۲

چکیده

برنامه ریزی مدارس یکی از دغدغه‌های مدیران و حتی دبیران در ابتدای سال تحصیلی است. در مقاله‌ی حاضر با استفاده از برنامه ریزی ریاضی این مساله مدل شده است. به دلیل تعداد بسیار بالای متغیرهای ۰ و ۱ مساله، یک مساله NP-complete است و مدل سازی عادی چندان عملی نخواهد بود. سه تکه (تجزیه) کردن مساله می تواند راه حل مناسبی برای غلبه بر این مشکل باشد. در تکه (فاز) اول این مدل تعیین می شود که هر دبیر در چه روزهایی باید در این مدرسه باشد و در فاز دوم تعیین می شود که دبیر در هر یک از این روزهای تعیین شده چه درسی را باید به چه کلاسی بدهد و در فاز سوم تعیین می شود که این دروس تعیین شده در هر روز برای هر کلاس در چه ساعتی باید توسط دبیر مربوط تدریس شود.

کلمات کلیدی: جدول زمانی، برنامه ریزی صفر و یک، مدل سازی ریاضی، تجزیه.

۱ مقدمه

برنامه ریزی دبیران در مدارس یعنی مشخص کردن این که هر دبیر در چه روزی و در چه ساعتی چه درسی را به چه کلاسی بدهد به طوری که هر دبیر درسی را که تبحر دارد درس دهد [۱] و هم چنین زمان مورد نیاز هر کلاس برای هر درس تامین و همه ی ساعت کار دبیران در یک مدرسه استفاده شود. حل این مساله ی پیچیده به صورت دستی مشکلات زیر را دارد:

۱. برنامه ریزی مدارس یکی از مشکلات ابتدای سال تحصیلی برای مدیران است. برنامه ریزی یک مدرسه نه چندان بزرگ چیزی حدود یک هفته از وقت یک فرد خبره را می گیرد.
۲. علاوه بر این جواب حاصل شده به دلایلی هم چون اشتباهات فردی و... لزوماً خوب نیست.
۳. گاهی اوقات در جواب حاصل شده به تغییر بی مورد روزهای مطلوب دبیران می انجامد.
۴. برخورد ناعادلانه با مدارس:

* عهده دار مکاتبات

آدرس الکترونیکی: vahidi@iust.ac.ir

مدارس معتبرتر به راحتی و انعطاف بیشتری برنامه خود را تعیین می‌کنند و به علت اعتبار خود، دبیران را مجبور می‌نمایند و یا دبیران ترجیح می‌دهند برنامه این مدارس را همان‌گونه که مدیر این مدارس به آن‌ها می‌گویند بپذیرند و روزهای مورد نظر مدیران را ارایه دهند و این باعث می‌شود برنامه مدارس پایین شهر و با اعتبار کمتر را تغییر دهند و برنامه این مدارس به سختی بپذیرند.

۵. چون در سازمان‌دهی دبیران رشته‌های مختلف به ترتیب سازمان‌دهی می‌شوند؛ ترجیحات دبیرانی که ابتدا سازمان‌دهی می‌گردند بدون دلیل بر ترجیحات دبیران بعدی ارجحیت می‌یابد.

در صورت انجام این کار به صورت کامپیوتری علاوه بر حل مشکلات مذکور سالانه حدوداً ۳۰۰ میلیون تومان از هزینه‌های آموزش و پرورش در کل کشور کاسته می‌شود.

در این پژوهش با استفاده از برنامه‌ریزی ریاضی مسأله مدل شده است. این مدل نوع گسترش‌یافته‌ای از مدل تخصیص است [۲] و برای مسایل جداول زمانی مانند جدول زمانی دروس دانشگاه و یا زمان‌بندی امتحانات نیز می‌تواند مورد استفاده قرار گیرد. همچنین این مدل شباهت بسیاری به برنامه‌ی سیستم‌های توزیع محصولات فاسدشدنی مانند لبنیات دارد. به همین دلیل می‌توان از کلیت چنین مدلی برای حل این مسأله نیز که یکی از مسایل اساسی در سیستم‌های توزیعی است؛ استفاده کرد [۳].

در پروژه حاضر با استفاده از برنامه‌ریزی ریاضی این مسأله مدل شده است. به دلیل تعداد بسیار بالای متغیرهای ۱۰ و ۱ مسأله، یک مسأله NP-complete می‌باشد [۴]. سه تکه کردن مسأله می‌تواند راه‌حل مناسبی برای غلبه بر این مشکل باشد. در تکه (فاز) اول این مدل تعیین می‌شود که هر دبیر در چه روزهایی باید در این مدرسه باشد و در فاز دوم تعیین می‌شود که دبیر در هر یک از این روزهای تعیین شده چه درسی را به چه کلاسی باید بدهد و در فاز سوم تعیین می‌شود که این دروس تعیین شده در هر روز برای هر کلاس در چه ساعتی باید توسط دبیر مربوط تدریس شود. علاوه بر این مدل جامعی برای حل فاز اول این مسأله برای کل مدارس یک شهر ارایه شده است.

برای وارد کردن این مدل در کامپیوتر برنامه GAMS انتخاب شده زیرا با برنامه Access به راحتی ارتباط برقرار می‌کند و در قسمت‌هایی از کار لازم است که اطلاعات ورودی از Access خوانده و اطلاعات خروجی به Access داده شود. همچنین این برنامه توان بالایی برای حل برنامه‌ریزی صفر و یک دارد.

در این مقاله، یک دبیرستان (روزانه ۶ ساعت) با ۱۵ کلاس با ۱۵ درس در هر کلاس، ۳۰ دبیر و ۶ روز در هفته در نظر می‌گیریم و مدل مربوط به آن را ارایه می‌دهیم.

۲ مراحل حل

فاز اول

در این فاز تعیین می‌کنیم هر دبیر در چه روزی در مدرسه است. متغیر صفر و یک X_{ij} مشخص می‌کند که آیا دبیر i در روز j در مدرسه کلاس دارد یا خیر. و متغیر صفر و یک X'_{ij} مشخص می‌کند که آیا دبیر i در روز j در مدرسه کلاس دارد یا خیر.

در مدرسه ساعات باقی مانده‌اش را دارد یا خیر (ساعت باقی مانده یعنی باقی مانده‌ی تقسیم ساعات کل تدریس یک دبیر در مدرسه تقسیم بر ۶. این تقسیم‌بندی در جهت رفاه حال دبیران و حضور کمتر آن‌ها انجام می‌گردد).
حدود ۳۰ دبیر (i) و ۶ روز داریم پس در کل حدود ۱۸۰ متغیر صفر و یک X_{ij} و ۷۰ متغیر صفر و یک X'_{ij} در این فاز خواهیم داشت.

P_i : اولویت دبیر i ام (که به صورت نسبی و با تکنیک‌هایی نسبت به سایر دبیران تعیین می‌شود).

P_{ij} : ترجیح دبیر i ام برای حضور در روز j ام.

بنابراین هر چه P_i و P_{ij} بیشتر شوند در نتیجه مضرب این دو بیشتر می‌شود بنابراین احتمال یک شدن X_{ij} با توجه به نوع تابع هدف مساله بیشتر می‌شود.

$$\text{Max } \sum_i \sum_j P_i P_{ij} (X_{ij} + m_i * X'_{ij})$$

تابع هدف

m_i برابر است با ساعت باقی‌مانده‌ی دبیر i ام تقسیم بر کل ساعت یک روز (۶ ساعت). مثلاً اگر دبیر i ام کلاً ۱۷ ساعت کلاس داشته باشد $m_i = \frac{5}{6}$ خواهد باشد).

محدودیت‌ها

رابطه‌ی ۱ یعنی هر دبیر به تعداد روزهای کاری کاملش (خارج قسمت تقسیم کل ساعت او بر ۶) در مدرسه باشد.

$$\sum_j X_{ij} = d_i \quad \forall i$$

رابطه‌ی ۲ یعنی دبیر i ام یک و فقط یک روز ناکامل (روزی که ساعات باقی‌مانده‌اش را در آن روز تدریس می‌کند) داشته باشد.

$$\sum_j X'_{ij} = 1 \quad \forall i'$$

رابطه‌ی ۱ و ۲ سبب می‌شود هر دبیر به تعداد کل ساعتش در مدرسه کلاس داشته باشد.

رابطه‌ی ۳ بیانگر رابطه‌ی بین دو دسته متغیر است و به این معنی است که هر روز برای هر دبیر هم روز کامل و هم روز ناکامل نباشد.

$$X_{ij} + X'_{ij} \leq 1 \quad \forall i, j$$

رابطه‌ی ۴ بیانگر این است که تعداد دبیرهای کامل هر روز به تعداد کلاس‌های مدرسه باشد تا دبیرهای مورد نیاز هر روز تامین شود.

تعداد کلاس‌های این مدرسه m:

$$\sum_i (X_{ij} + m_i * X'_{ij}) = n$$

فاز دوم

با تخصیص دبیران به روزهای مختلف در این فاز به تخصیص دبیران به هر درس هر کلاس می‌پردازیم. برای این منظور متغیرهایی را برای فاز ۲ تعریف می‌کنیم که متغیرهای متناسب آن‌ها در فاز اول ۱ شده است. متغیر صفر و یک X_{ilc} مشخص می‌کند که آیا دبیر i ام درس l کلاس c را دارد یا خیر. درس‌ها را به صورت زوج مرتب (c, l) در نظر می‌گیریم در این حالت $l = 2$ در $(2, 3)$ با $l = 2$ در $(2, 4)$ فرق خواهد داشت.

حدوداً ۳۰ دبیر (i) داریم که هر یک به طور میانگین ۲ درس را می‌توانند درس دهند (l) و حدود ۱۵ کلاس داریم (c). پس این دسته متغیر حدود ۹۰۰ متغیر را شامل می‌شود. متغیر صفر و یک X_{ijlc} مشخص می‌کند که آیا دبیر i ام در روز j ام یک زنگ درس l را با کلاس c ام دارد یا خیر. حدوداً ۳۰ دبیر (i) داریم که هر یک به طور میانگین ۲ درس را می‌توانند درس دهند (l) و معمولاً ۳ روز از هفته را به این مدرسه می‌آید (j) و حدوداً به طور میانگین می‌تواند به ۱۰ کلاس نیز درس دهد (c)، پس این دسته متغیر حدود ۱۸۰۰ متغیر را شامل می‌شود و کل متغیرها ۲۷۰۰ متغیر خواهد بود.

تابع هدف

تابع هدف در این فاز حداکثر کردن توانایی دبیران برای تدریس در رشته‌ها و پایه‌های مورد نظر و یکنواخت کردن سختی روزهای مختلف:

$$\text{Max} \sum_c \sum_l \sum_i (d_{ilc} X_{ilc}) - c * D$$

محدودیت‌ها

d_{ilc} : میزان توانایی دبیر i ام برای تدریس درس l ام در کلاس c ام (با توجه به رشته و پایه)
 D یک متغیر است که نشان‌دهنده ماکزیمم سختی دروس روزهای هفته می‌باشد و c نیز ضریب وزن دادن به آن در تابع هدف است. این مولفه‌ی تابع هدف سبب می‌شود تا سختی بیشینه کاهش یابد و سختی دروس در روزهای مختلف به طور یکنواخت پراکنده شود. D_{ilc} نیز سختی درس l ام در کلاس c ام است:

$$\begin{aligned} \sum_i \sum_l X_{ijlc} * D_{ilc} &\leq D & \forall j, c \\ \sum_i \left[X_{ilc} * h_l - \left(\sum_j X_{ijlc} \right) \right] &= 0 & \forall (l, c) \\ \sum_i X_{ilc} &= 1 & \forall (l, c) \end{aligned}$$

رابطه‌ی ۵ یعنی هر درس را برای هر کلاس (هر (l, c)) یک دبیر درس دهد یعنی مثلاً ریاضی کلاس ۱۱۰ را فقط آقای فلانی درس دهد نه این که دو نفر این درس را به یک کلاس خاص بدهند.

رابطه‌ی ۵ و ۶ بیانگر رابطه‌ی بین دو دسته متغیر مساله است و هم‌چنین سبب می‌شود هر درس هر کلاس را یک نفر درس دهد و هم‌چنین نیاز هر کلاس به ساعات هر درس را نیز مرتفع می‌کند و سبب می‌شود دبیر h_l ساعت صرف این درس کند:

$$\sum_i \sum_l X_{ijlc} = 3 \quad \forall j, c$$

رابطه‌ی ۷ یعنی در هر روز برای هر کلاس ۳ (به تعداد زنگ‌ها) دبیر باشد:

$$\sum_i X_{ijlc} = 1 \quad \forall j, l, c$$

$$\sum_c \sum_l X_{ijlc} = h_{ij} \quad \forall (i, j)$$

رابطه‌ی ۸ بیانگر این است که هر دبیر در هر روز به تعداد تعیین شده‌اش در فاز قبل (h_{ij}) درس دهد. (در نرم‌افزارها مانند GAMS می‌توان برای کاهش حجم مساله این محدودیت را فقط بر روی زهای ممکن برای هر دبیر تعریف کرد. این محدودیت هم‌چنین سبب می‌شود هر دبیر در هر زنگ حداکثر در یک کلاس باشد و هم‌چنین هر کلاس در هر زنگ دبیر داشته باشد.

در صورت وجود درس تک‌زنگ برای این دروس به جای X_{ijlc} ، $(X_{ijlc} + \frac{1}{4} X'_{ijlc})$ قرار می‌دهیم که X'_{ijlc} متغیر صفر و یکی است که نشان می‌دهد ساعت تک‌زنگ این درس در چه روزی است و نیز باید محدودیت زیر را قرار دهیم تا این متغیر فقط برای یک روز و یک دبیر یک شود:

$$\sum_j \sum_i X'_{ijlc} = 1 \quad \forall j, i$$

فاز سوم

در این مرحله پس از تعیین این که هر دبیر چه درسی و چند ساعت و در چه روزی به چه کلاسی درس دهد؛ باید تعیین کنیم که این درس را در چه زنگی درس دهد. در این مرحله چون نیازهای کلاس‌ها به دروس و روابط بین ساعات مختلف یک درس در فاز قبل در نظر گرفته شده است؛ می‌توانیم روزها را به طور مجزا برنامه‌ریزی کنیم. X_{iklc} متغیری است که نشان می‌دهد دبیر i ام در کلاس c ام یک زنگ درس l را در زنگ k دارد یا نه.

البته این متغیر را برای متغیرهایی که متغیر متناسب با آن یک شود؛ تعریف می‌کنیم.

برای هر درس تک‌زنگ X'_{ijkc} را مشابه تعریف گفته شده در فاز قبل تعریف می‌کنیم و به جای X_{ijkc} قرار

$$\text{می‌دهیم } (X_{ijkc} + \frac{1}{4} X'_{ijkc}).$$

در کل این فاز حدود ۱۵۰ متغیر صفر و یک خواهد داشت.

تابع هدف

تابع هدف در این فاز حداکثر کردن مطلوبیت دروس برای قرار گرفتن در ساعت‌ها مختلف روز مد نظر (u_{ijklc}) است:

$$\text{Max} \sum_i \sum_k \sum_l \sum_c (u_{ijklc} * X_{iklc}) - c * D$$

محدودیت‌ها

به علت مساوی بودن عرضه و تقاضای ساعت در هر زنگ یک نفر یا دو نفر تک‌زنگ خواهیم داشت:

$$\sum_c \sum_l X_{ilkc} \leq 1 \quad \forall i, k$$

محدودیت ۹ بیانگر این است که هر نفر در هر زنگ فقط در یک کلاس باشد. به علت مساوی بودن عرضه و تقاضای ساعت، در هر زنگ یک نفر یا دو نفر تک‌زنگ خواهیم داشت و محدودیت ۹ را می‌توان حذف کرد:

$$\sum_i \sum_l X_{ilkc} = 1 \quad \forall c, k$$

محدودیت ۱۰ بیانگر این است که هر کلاس در هر زنگ دبیر داشته باشد:

$$\sum_k \sum_c X_{iklc} = h_{ilc} \quad \forall i, l, c$$

h_{ilc} تعداد زنگ دبیر l ام برای درس l ام کلاس c ام است که در فاز قبل مشخص شده است.

۳ نتیجه‌گیری

۳-۱ بررسی امکان آزادسازی خطی مسأله

با توجه به این که باید مدل totally unimodular باشد تا چنین حالتی رخ دهد به بررسی این ویژگی در مدل‌ها می‌پردازیم [۵].

فاز یک ماتریس ضرایبش به شکل زیر می‌شود که درمیان آن لزوماً ۱، -۱ و ۰ نمی‌گردد.

$$\begin{pmatrix} 1 & 1 & \dots & 1 & 0 & \dots & \dots & 0 \\ 1 & 1 & \dots & 1 & \vdots & \ddots & & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & & \ddots & \vdots \\ 1 & 1 & \dots & 1 & 0 & \dots & \dots & 0 \\ 0 & \dots & \dots & 0 & 1 & 1 & \dots & 1 \\ \vdots & \ddots & & \vdots & 1 & 1 & \dots & 1 \\ \vdots & & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & 0 & 1 & 1 & \dots & 1 \\ 1 & & & \dots & & & & 1 \\ \vdots & & & & & & & \vdots \\ 1 & & & \dots & & & & 1 \\ 1 & & & & & & & 1 \\ & & & & & & m_i & \\ & & & & & & & 1 \end{pmatrix}$$

شکل ۱. ماتریس بررسی امکان آزادسازی خطی

به علت وجود ضریب m_i در مدل totally unimodular نیست. فاز دو نیز دترمینان آن لزوماً ۱-، ۰ و ۱ نمی شود؛ به علت وجود ضریب D_1 و h_1 در مدل totally unimodular نیست. دترمینان فاز سه نیز به علت وجود ضریب ۰/۵ برای متغیرهای X'_{ijk} totally unimodular نمی باشد.

۲-۳ اگر مساله به جواب نرسید

۱. می توان محدودیت های غیر اصلی را غیر فعال کرد. (به ترتیب اولویت محدودیت ها تا جایی که به جواب برسیم).
۲. اگر این روش جواب نداد؛ مجبوریم تعداد زنگ ها را به جای ۳، ۶ بگیریم.

۳-۳ محدودیت های فرعی

محدودیت های دیگری نیز در برنامه ریزی مدارس وجود دارد اما این محدودیت ها، محدودیت های اصلی این مساله نیستند و از مدرسه ای به مدرسه ای دیگر فرق می کنند که ما در این جا تعدادی از مهم ترین های آن ها را ذکر کرده؛ مدل مربوط به آن ها را ارایه خواهیم کرد.

- مطلوبیت های دیگر مساله را هم می توان به صورت تابع هدف و یا آرمان در فاز مربوط به آن وارد نمود.
- اضافه کاری: برای افزودن ساعات اضافه کاری دبیران به برنامه باید ارجحیت (ضریب P_i) کمتری برای دبیران اضافه کاری در نظر گرفت تا ابتدا ساعات مورد نظر دبیران موظفی پر شود.

اگر دبیری داشتیم که در مدرسه‌ای هم موظفی و هم اضافه کاری داشت می‌توان آن را دو دبیر مجزا در نظر گرفت و محدودیتی اضافه کرد که این دو دبیر مجزاً، مجزا بتوانند یک ساعت از درس را یکی و یک ساعت دیگرش را دیگری درس دهد. البته بسیار کم پیش می‌آید که چنین حالتی رخ دهد.

- درس‌هایی که کمبود دبیر در آن در شهر است: به دبیران مربوط به این دروس ضرایب P_i بزرگ‌تری اختصاص می‌دهیم.
- محدودیت مکانی بعضی از دروس مانند کارگاه، آزمایشگاه، سایت و ورزش: مثلاً سایت:

$$\sum_i \sum_c (X_{ijk} \times X_{ilc}) = n_{site} \quad \forall j, k$$

n_{site} تعداد سایت‌های مدرسه است.

- تداخل شیفت صبح و بعد از ظهر یک مدرسه: برای خالی شدن کلاس‌ها باید تعدادی از کلاس‌ها را به سایت، زمین ورزش، آزمایشگاه و یا کارگاه فرستاد.
- فرض کنید شماره این ۴ درس از l_1 تا l_p باشد:

$$\sum_i \sum_{l_1}^{l_p} (X_{ij'c} \times X_{ilc'}) = 1 \quad \forall c', j'$$

j' روزهایی که این حالت وجود دارد و c' کلاس‌هایی که تداخل دارند.

عبارت داخل پرانتز یعنی دبیر i با کلاس c درس l را در زمان jk دارد.

- دبیری کلاس، پایه، درس و یا زمان خاصی را نمی‌خواهد: مثلاً درس خاصی را نمی‌خواهد:

$$\sum_c X_{i_1 l_1 c} = 0$$

۳-۴ مزایای این روش به مدل عادی زمان‌بندی برای این مساله

اگر بخواهیم این مساله را به روش تخصیص معمولی حل کنیم حدود ۱۲۰۰۰۰ متغیر داریم که تعداد بسیار زیادی است.

$$120000 / 2700 = 45$$

تعداد متغیرهای این مساله ۴۵ حداکثر (فاز ۲) برابر بیشتر است. X_{ijklc} متغیری است که نشان می‌دهد دبیر am در روز am یک زنگ درس l را در زنگ k دارد یا نه و حتی اگر با استفاده از روش این مساله به ازای هر l فقط دو درس در نظر بگیرید در کل حدود ۱۶۰۰۰ متغیر خواهیم داشت.

تعداد متغیرهای این مساله حدود ۶ برابر بیشتر است.

$$16000 / 2700 \approx 6$$

منابع

- [۱] سواد کوهی، عبدالعظیم، طراحی مدل برنامه‌ریزی آموزشی، پایان‌نامه کارشناسی ارشد، دانشگاه امیرکبیر.
- [2] Bazaraa, M. S., Jarvis, J. J., Sherali, H. D., (2004). Linear Programming and Network Flows. Wiley-Interscience.
- [3] Winston, W. L., Goldberg, J. B., (2004). Operations research: applications and algorithms. Thomson Brooks/Cole.
- [4] Handbooks in operations research and management science contents of the previous volume. (1990). In D. P. Heyman & M. J. Sobel (Eds.), Handbooks in Operations Research and Management Science (Vol. Volume 2, pp. 725): Elsevier.
- [5] Murty, K. G., (1983). Linear programming. Wiley.